Learning and Earning Under Noise and Uncertainty

Su Jia

Tepper School of Business

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Algorithms, Combinatorics and Optimization

Thesis Committee:
R. Ravi (Chair)
Andrew A. Li
Alan Scheller-Wolf
Sridhar Tayur

Contents

List of Tables

Acknowledgments.

The first and most important acknowledgment belongs to my advisors R. Ravi and Andrew Li. Despite the considerable demands on their time, they still managed to spend considerable amount of time on me. I also want to thank all professors in our Operations Research department and my fellow students in the OR group and ACO program for their friendship. I particularly want to thank everyone who coauthored with me during my PhD career in CMU: Ian Anderson, Paul Duff, Kyra Gan, Jeremy Karp, Andrew Li, Viswanath Nagarajan, Fatemeh Navidi, Nishant Oli, R Ravi and Sridar Tayur. I also want to thank Alan Scheller-Wolf for his service on my thesis committee. I would also like to deeply thank Joseph Mitchell and Jie Gao in Stony Brook University for serving as the very first mentors in my early research career, and providing me with solid research training in algorithms so that I could get on track quickly in CMU. Finally, I conclude by thanking my family, especially my parents, for their support and understanding.

Chapter I Introduction

Sequential decision-making under uncertainty is central to a range of operations and marketing problems. In the face of an unknown environment, decision-maker needs to strike a balance between learning the known environment ("learning") and selecting nearly optimal decisions ("earning"). For instance, consider pricing a new product. If the retailer had full information about the demand at every price level, then she could determine the revenue-maximizing price for the good. However, such information about the demand curve is typically not available in practice, so the seller needs to experiment with different prices to gain information about the demand curve, and then exploit this information to offer a near-optimal selling price.

The trade-off between learning and optimization can be modelled as the Multi-Armed Bandits (MAB) problem, or more generally, online learning, which has attracted significant attention from a range of communities in recent years, including machine learning, operations research and marketing. While most of the fundamental problems in this area have been theoretically well-understood, these algorithms have been rarely deployed in practice. In contrast, while marketing research on sequential decision making has been focused on the practical side, their results are usually empirical and lacking of rigorous analysis.

This thesis serves as a preliminary step towards filling this gap. We will consider *practical* sequential decision-making problems arising from some of the most fundamental areas in marketing, including survey design, pricing and content recommendation, and provide *theoretical* insights via provable performance guarantees.

I.1. Optimal Decision Tree Problem Under Noisy Outcomes

From Spotify to Netflix and Amazon, we are surrounded by extreme personalization every day. Consumers have come to expect that same level of personalization from companies of all sizes. Investing in personalization efforts to build relationships and create better experiences can pay off with serious rewards for brands. And in a world where the vast majority of companies are focused on improving personalization, companies that do not prioritize creating a tailored experience run the risk of getting left behind.

One approach to personalized service for new users is by classifying users into typical user-types and then identifying the user-type based on their responses to survey questions. The problem of designing efficient surveys is accurately modelled by the Optimal Decision Tree (ODT) Problem, where decision maker needs to perform a sequence of tests to identify an unknown hypothesis drawn from a known distribution. The basic version of the ODT problem has been widely studied for decades and an asymptotically best-possible approximation algorithm has been devised. However in practice, the test outcomes are usually noisy, caused, for example, by the user heterogeneity within each group, rendering these algorithms inapplicable to real world problems.

This motivates us to study a generalization of the ODT problem where the outcomes are contaminated by persistent noise, that is, the outcomes of certain tests may be flipped, but remains the same each time the test is performed. More generally, we introduce a problem, Submodular Function Ranking with Noise, that generalizes the above problem. Despite the extensive literature on the ODT problem and a closely related machine learning field called *active learning*, little is known about the noisy version. There are two main reasons. First, the persistence of noise disables most of the statistical learning tools such as concentration bounds. Secondly, the structure of the optimal solution becomes significantly more complicated under noisy outcomes, posing substantial challenge for theoretical analysis, especially in terms of approximation ratio.

In Chapter II, we design new approximation algorithms for both the non-adaptive setting, where the test sequence must be fixed *a-priori*, and the adaptive setting where the test sequence depends on the outcomes of prior tests. Our new approximation algorithms provide guarantees that are nearly best-possible and work for the general case of a large number of noisy outcomes per test or per hypothesis where the performance degrades smoothly with this number. Moreover, numerical evaluations show that despite our theoretical logarithmic approximation guarantees, our methods give solutions with cost very close to the information theoretic minimum, demonstrating the effectiveness of our methods.

This chapter is based on Jia et al. (2019), a joint work with Fatemeh Navidi, Viswanath Nagarajan and R. Ravi. Further, our joint work (Gan et al. (2021)) with Kyra Gan, Andrew Li and Sridhar Tayur extended the results to the error-budgeted version based upon the techniques developed in this chapter, and received the *2021 Pierskalla Best Paper Award in Healthcare Applications* for its novel application in cancer research.

I.2. Markdown Pricing Under Unknown Demand

Dynamic pricing under unknown demand has been theoretically well-understood, usually under the framework of multi-armed bandits. But in practice, these bandit-based policies are rarely deployed by real-world retailers, largely because oscillating prices may cause customer dissatisfaction. For example, Luca and Reshef (2021) discovered that a “1% price increase (in menu prices) leads to a 3% to 5% decrease in online ratings on average”.

This motivates us to consider dynamic pricing in Chapter III and IV under the monotonicity constraint, that is, the prices must be non-increasing. While both markdown pricing under *known* demand and *unconstrained* pricing under unknown demand have been well-understood, little is known for markdown pricing under unknown demand. In particular, the following basic questions remains open prior to this work.

What is the optimal regret bound for markdown pricing? In particular, how does this bound compare with the known bounds for unconstrained pricing?

For instance, under the Lipschitz assumption, Kleinberg (2005) showed an asymptoticly best possible $\Theta(T^{2/3})$ regret bound for unconstrained pricing in T rounds. Can we show that the optimal regret bound for markdown pricing is asymptoticly higher than $T^{\frac{2}{3}}$?

We provide a **complete settlement** of this fundamental question in Chapter III and IV. More precisely, we present optimal regret bounds for markdown pricing, under various assumptions, from the most agnostic setting where only the minimal assumptions are imposed for deriving meaningful guarantees, to the most fine-grained setting where the demand curve is assumed to come from certain class of well-behaved functions. Furthermore, in almost every regime, our tight bound is asymptoticly higher than the known bounds under the same assumptions, highlighting the extra complexity introduced by the monotonicity constraint.

Finally, we also investigate various extensions of this problem, including the scenario where the monotonicity constraint can be relaxed at a certain cost. This work also opens up a wealth of other related new directions for future study. These two chapters are both joint with Andrew Li and R. Ravi.

I.3. Short-Lived High-Volume Bandits: Algorithms and Field Experiment

In the final chapter, we consider problem of recommending short-lived contents to users. There has been a long history where online platforms leverage the scale of data, especially related to user attention, to make better decisions for newly-arriving products or content. By and large, recommendation tasks can be classified into four categories based on the *lifetime* and *volume* of contents generated. For persistent (long-lived) content, the problem is arguably straightforward: spend a negligible amount of time collecting sufficient data in the form of user feedback, and then apply suitable offline predictive model, which might range in sophistication from a basic collaborative filtering algorithms to, nowadays, deep neural networks (DNNs). For example, recently YouTube deployed a recommender system comprised of two deep neural networks: one for candidate generation and one for ranking.

Orthogonal to content lifetime, when there is a *low volume* of content relative to the number of users, the problem is similarly well-understood: dedicated exploration methods (e.g. A/B testing) are sufficient for finding the right segments of users for which the content is most appealing. LinkedIn runs over 400 concurrent experiments per day to compare different designs of their website, with the goal of, for example, encouraging the users to better establish their personal profile, or increasing the subscriptions to LinkedIn Premium.

Naturally then, the most challenging settings are where the content to be recommended is *short-lived* and *high-volume*. Such settings arise, for example, in content aggregation platforms (e.g. Apple News) and platforms with content that is entirely user-generated (e.g. TikTok). In these settings, both of the previous approaches are prone to failure: offline predictive algorithms do not receive enough data on individual content to achieve meaningful accuracy due to the short lifetime, and dedicated exploration methods are ill-suited to high volume.

The question then is, how should an online platform decide what content to display to each user? In addition to the well-known “learn-and-earn” trade-off in MAB, the online platform needs to resolve an additional concern: the balance between the exploration of newly arriving and older contents. We propose a simple bandit-based approach for recommending short-lived content, which we show to have nearly-optimal performance guarantee.

Further, we implemented this policy in a live field experiment with Glance, a leading lockscreen content platform in India, which faces exactly this challenge. Over the course of two weeks, our policy achieved a 12% improvement in conversions rates, relative to the neural network based control policy. This chapter is based on our joint work with Nishant Oli, Andrew Li, R. Ravi, Paul Duff and Ian Anderson.

Chapter II Optimal Decision Tree and Submodular Ranking with Noisy Outcomes

A fundamental task in active learning involves performing a sequence of tests to identify an unknown hypothesis that is drawn from a known distribution. This problem, known as optimal decision tree induction, has been widely studied for decades and the asymptotically best-possible approximation algorithm has been devised for it. We study a generalization where certain test outcomes are noisy, even in the more general case when the noise is persistent, i.e., repeating a test gives the same noisy output. More generally, we introduce a problem, Submodular Function Ranking with Noise, that generalizes the above problem. We design new approximation algorithms for both the non-adaptive setting, where the test sequence must be fixed *a-priori*, and the adaptive setting where the test sequence depends on the outcomes of prior tests. Previous work in the area assumed at most a logarithmic number of noisy outcomes per hypothesis and provided approximation ratios that depended on parameters such as the minimum probability of a hypothesis. Our new approximation algorithms provide guarantees that are nearly best-possible and work for the general case of a large number of noisy outcomes per test or per hypothesis where the performance degrades smoothly with this number. In fact, our results hold in a significantly more general setting, where the goal is to cover stochastic submodular functions.

We numerically evaluate the performance of our algorithms on two natural applications with noise: toxic chemical identification and active learning of linear classifiers. Despite our theoretical logarithmic approximation guarantees, our methods give solutions with cost very close to the information theoretic minimum, demonstrating the effectiveness of our methods.

II.1. Introduction

The classic Optimal Decision Tree (ODT) problem involves identifying an initially unknown *hypothesis* h that is drawn from a known probability distribution over a set of hypotheses. We can perform *tests* in order to distinguish between these hypotheses. Each test produces a binary outcome (positive or negative) and the precise outcome of each test-hypothesis pair is known beforehand, and thus an instance of ODT can be viewed as a ± 1-valued matrix M with the tests as rows and hypotheses as columns. The goal is to identify the true hypothesis h using the fewest tests.

As a motivating application, consider the following task in medical diagnosis detailed in Loveland (1985). A doctor needs to diagnose a patient's disease by performing tests. Given an *a priori* probability distribution over possible diseases, what sequence of tests should the doctor perform to identify the disease as quickly as possible? Another application is in active learning (e.g. Dasgupta (2005)). Given a set of data points, one wants to learn a classifier that labels the points correctly as positive and negative. There is a set of m possible classifiers which is assumed to contain the true classifier. In the Bayesian setting, which we consider, the true classifier is drawn from some known probability distribution. The goal is to identify the true classifier by querying labels at the minimum number of points in expectation (over the prior distribution). Other applications include entity identification in databases (Chakaravarthy et al. (2011)) and experimental design to choose the most accurate theory among competing candidates (Golovin et al. (2010)).

Despite the considerable literature on the classic ODT problem, an important issue that is not considered is that of unknown or noisy outcomes. In fact, our research was motivated by a data-set involving toxic chemical identification where the outcomes of many hypothesis-test pairs are stated as unknown (see Section III.6 for details). While prior work incorporating noise in ODT, for example Golovin et al. (2010), was restricted to settings with very few noisy outcomes, in this paper, we design approximation algorithms for the noisy optimal decision tree problem in full generality.

Specifically, we generalize the ODT problem to allow unknown/noisy entries (denoted by "$*$") in the test-hypothesis matrix M, to obtain the *Optimal Decision Tree with Noise* (ODTN) problem, in which the outcome of each noisy entry in the test-hypothesis matrix M is a random ± 1 value, independent of other noisy entries. More precisely, if the entry $M_{t,h} = *$ (for hypothesis h and test t) and the realized hypothesis is h, then the outcome of t will be a random ± 1 value. We will assume for simplicity that each noisy outcome is ± 1 with uniform probability, though our results extend directly to the case where each noisy outcome has a different probability. We consider the standard *persistent* noise model, where repeating the same test always produces the same outcome. Note that this model is more general than the non-persistent noise (where repeating a noisy test leads to "fresh" independent ± 1 outcomes), since one may create copies of tests and hypotheses to reduce to the persistent noise model.

We consider both non-adaptive policies, where the test sequence is fixed upfront, and adaptive policies, where the test sequence is built incrementally and depends on observed test outcomes. Evidently, adaptive policies perform at least as well as non-adaptive ones. Indeed, there exists instances where the relative gap between the best adaptive and non-adaptive policies is very large (see for example, Dasgupta (2005)). However, non-adaptive policies are very simple to implement, requiring minimal incremental computation, and may be preferred in time-sensitive applications.

In fact, our results hold in a significantly more general setting, where the goal is to cover *stochastic* submodular functions. In the absence of noisy outcomes, the non-adaptive and adaptive versions of this problem were studied by by Azar and Gamzu (2011) and Navidi et al. (2020). Other than the ODT problem, this submodular setting captures a number of applications such as multiple-intent search ranking, decision region determination and correlated knapsack cover: see Navidi et al. (2020) for details. Our work is the first to handle noisy outcomes in all these applications.

II.2. Contributions

We derive most of our results for the ODTN problem as corollaries of a more general problem, Submodular Function Ranking with Noisy Outcomes, which is a natural extension of the Submodular Function Ranking problem, introduced by Azar and Gamzu (2011). We first state our results before formally defining this problem in Section II.4.3.

First, we obtain an $O(\log\frac{1}{\varepsilon})$-approximation algorithm (see Theorem 3) for *Non-Adaptive* Submodular Function Ranking with noisy outcomes (SFRN) where ε is a separability parameter of the underlying submodular functions. As a special case, for the ODTN (both adaptive and non-adaptive) problem, we consider submodular functions with separability $\varepsilon=\frac{1}{m}$, so the above result immediately implies an $O(\log m)$-approximation for non-adaptive ODTN. This bound is the best possible (up to constant factors) even in the noiseless case, assuming $P\neq NP$.

As our second contribution, we obtain an $O(\min\{c\log|\Omega|,r\}+\log\frac{m}{\varepsilon})$-approximation (Theorem 7) algorithm for *Adaptive* Submodular Ranking with noisy outcomes (ASRN), which implies an $O(\min\{c,r\}+\log m)$ bound for ODTN by setting $\varepsilon=\frac{1}{m}$, where Ω is the set of random outcomes we may observe when selecting elements. The term $\min\{c\log|\Omega|,r\}$ corresponds to the "noise sparsity" of the instance (see Section II.4 for formal definitions). For the ODTN problem, c (resp. r) is the maximum number of noisy outcomes in each column (resp. row) of the test-hypothesis matrix M. In the noiseless case, $c=r=0$ and our result matches the best approximation ratio for the ODT and the Adaptive Submodular Ranking problem (Navidi et al. (2020)). In the noisy case, our performance guarantee degrades smoothly with the noise sparsity. For example, we obtain a logarithmic approximation ratio (which is the best possible) as long as the number of noisy outcomes in each row or column is at most logarithmic. For ODTN, Golovin et al. (2010) obtained an $O(\log^2 \frac{1}{p_{min}})$-approximation algorithm which is polynomial-time only when $c = O(\log m)$; here $p_{min} \leq \frac{1}{m}$ is the minimum probability of any hypothesis. Our result improves this result in that (i) the running time is polynomial irrespective of the number of noisy outcomes and (ii) the approximation ratio is better by at least one logarithmic factor.

While the above algorithm admits a nice approximation ratio when there are *few* noisy entries in each row or column of M, as our third contribution, we consider the other extreme, when each test has only a few *deterministic* entries (or equivalently, a *large* number of noisy outcomes). Here, we focus on the special case of ODTN. At first sight, higher noise seems to only render the problem more challenging, but somewhat surprisingly, we obtain a much *better* approximation ratio in this regime. Specifically, if the number of noisy outcomes in each test is at *least* $m - O(\sqrt{m})$, we obtain an approximation algorithm whose cost is $O(\log m)$ times the optimum and returns the target hypothesis with high probability. We establish this result by relating the cost to a *Stochastic Set Cover* instance, whose cost lower-bounds that of the ODTN instance.

Finally, we tested our algorithms on synthetic as well as a real dataset (arising in toxic chemical identification). We compared the empirical performance guarantee of our algorithms to an information-theoretic lower bound. The cost of the solution returned by our non-adaptive algorithm is typically within 50% of this lower bound, and typically within 20% for the adaptive algorithm, demonstrating the effective practical performance of our algorithms.

As a final remark, although in this work we will consider uniform distribution for noisy outcomes, our results extend directly to the case where each noisy outcome has a different probability of being ± 1. Suppose that the probability of every noisy outcome is between δ and $1 - \delta$. Then our results on ASRN continue to hold, irrespective of δ, and the result for the many-unknowns version holds with a slightly worse $O(\frac{1}{\delta} \log m)$ approximation ratio.

II.3. Related Work

The optimal decision tree problem (without noise) has been extensively studied for several decades: see Garey and Graham (1974), Hyafil and Rivest (1976/77), Loveland (1985), Arkin et al. (1998), Kosaraju et al. (1999), Adler and Heeringa (2008), Chakaravarthy et al. (2009), Gupta et al. (2017). The state-of-the-art result Gupta et al. (2017) is an $O(\log m)$-approximation, for instances with arbitrary probability distribution and costs. Chakaravarthy et al. (2011) also showed that ODT cannot be approximated to a factor better than $\Omega(\log m)$, unless P=NP.

The application of ODT to Bayesian active learning was formalized in Dasgupta (2005). There are also several results on the *statistical complexity* of active learning. e.g. Balcan et al. (2006), Hanneke (2007), Nowak (2009), where the focus is on proving bounds for structured hypothesis classes. In contrast, we consider arbitrary hypothesis classes and obtain *computationally efficient* policies with provable approximation bounds relative to the optimal (instance specific) policy. This approach is similar to that in Dasgupta (2005), Guillory and Bilmes (2009), Golovin and Krause (2011), Golovin et al. (2010), Cicalese et al. (2014), Javdani et al. (2014).

The noisy ODT problem was studied previously in Golovin et al. (2010). Using a connection to adaptive submodularity, Golovin and Krause (2011) obtained an $O(\log^2 \frac{1}{p_{min}})$-approximation algorithm for noisy ODT in the presence of very few noisy outcomes, where $p_{min} \leq \frac{1}{m}$ is the minimum probability of any hypothesis.[*] In particular, the running time of the algorithm in Golovin et al. (2010) is exponential in the number of noisy outcomes per hypothesis, which is polynomial only if this number is at most logarithmic in the number of hypotheses/tests. As noted earlier, our result improves both the running time (it is now polynomial for any number of noisy outcomes) and the approximation ratio. We note that an $O(\log m)$ approximation ratio (still only for very sparse noise) follows from work on the "equivalence class determination" problem by Cicalese et al. (2014). For this setting, our result is also an $O(\log m)$ approximation, but our algorithm is simpler. More importantly, ours is the first result that can handle *any* number of noisy outcomes.

Other variants of noisy ODT have also been considered, e.g. Naghshvar et al. (2012), Bellala et al. (2011), Chen et al. (2017), where the goal is to identify the correct hypothesis with at least some target probability. The theoretical results in Chen et al. (2017) provide "bicriteria" approximation bounds where the algorithm has a larger error probability than the optimal policy. Our setting is different because we enforce *zero* probability of error.

Many algorithms for ODT (including ours) rely on some underlying submodularity properties. We briefly survey some background results. In the basic Submodular Cover problem, we are given a set of elements and a submodular function f. The goal is to use the minimal number of elements to make the value of f reach certain threshold. Wolsey (1982) first considered this problem and proved that the natural greedy algorithm is a $(1 + \ln \frac{1}{\varepsilon})$-approximation algorithm, where ε is the minimal positive marginal increment of the function. As a natural generalization, in the Submodular

[*] The paper Golovin et al. (2010) states the approximation ratio as $O(\log \frac{1}{p_{min}})$ because it relied on an erroneous claim in Golovin and Krause (2011). The correct approximation ratio, based on Nan and Saligrama (2017), Golovin and Krause (2017), is $O(\log^2 \frac{1}{p_{min}})$.

Function Ranking problem we are given *multiple* submodular functions, and need to *sequentially* select elements so as to minimize the total cover time of those functions. Azar and Gamzu (2011) obtained an $O(\log \frac{1}{\epsilon})$-approximation algorithm for this problem, and Im et al. (2016) extended this result to also handle costs. More recently, Navidi et al. (2020) studied an adaptive version of the submodular ranking problem.

Finally, we note that there is also work on minimizing the *worst-case* (instead of average case) cost in ODT and active learning; see e.g., Moshkov (2010), Saettler et al. (2017), Guillory and Bilmes (2010, 2011). These results are incomparable to ours because we are interested in the average case, i.e. minimizing expected cost.

II.4. Preliminaries

II.4.1. Optimal Decision Tree with Noise In the Optimal Decision Tree with Noise (ODTN) problem, we are given a set of m possible *hypotheses* with a *prior* probability distribution $\{\pi_i\}_{i=1}^m$, from which an unknown hypothesis $\bar{i}$ is drawn. There is also a set $\mathcal{T}$ of n binary *tests*, each test $T \in \mathcal{T}$ associated with a 3-way partition T^+, T^-, T^* of $[m]$, where the outcome of test T is

- positive if $\bar{i} \in T^+$,
- negative if $\bar{i} \in T^-$, and
- positive or negative with probability $\frac{1}{2}$ each if $\bar{i} \in T^*$ (noisy outcomes).

We assume that conditioned on $\bar{i}$, each noisy outcome is independent. The outcomes for all test-hypothesis pairs can be summarized in a $\{1, -1, *\}$-valued $n \times m$ matrix M.

While we know the 3-way partition T^+, T^-, T^* for each test $T \in \mathcal{T}$ upfront, we are *not* aware of the actual outcomes for the noisy test-hypothesis pairs. It is assumed that the realized hypothesis $\bar{i}$ can be uniquely identified by performing all tests, regardless of the outcomes of $\star$-tests. This means that for every pair $i, j \in [m]$ of hypotheses, there is some test $T \in \mathcal{T}$ with $i \in T^+$ and $j \in T^-$ or vice-versa. The goal is to perform a sequence of tests to identify hypothesis $\bar{i}$ using the minimum *expected* number of tests, which will be formally defined soon. Note that the expectation is taken over both the prior distribution of $\bar{i}$ and the random outcomes of noisy tests for $\bar{i}$.

Types of Policies. A *non-adaptive* policy is specified by a permutation of tests denoting the order in which they will be tried until identification of the underlying hypothesis. The policy performs tests in this sequence and eliminates incompatible hypotheses until there is a unique compatible hypothesis (which is $\bar{i}$). Note that the number of tests performed under such a policy is still random as it depends on $\bar{i}$ and the outcomes of noisy tests.

An *adaptive* policy chooses tests incrementally, depending on prior test outcomes. The *state* of a policy is a tuple (E, d) where $E \subseteq \mathcal{T}$ is a subset of tests and $d \in \{\pm 1\}^E$ denotes the observed outcomes of the tests in E. An adaptive policy is specified by a mapping $\Phi : 2^{\mathcal{T} \times \{\pm 1\}} \to \mathcal{T}$ from states to tests, where $\Phi(E, d)$ is the next test to perform at state (E, d). Define the (random) cost $Cost(\Phi)$ of a policy Φ to be the number of tests performed until $\bar{i}$ is uniquely identified, i.e., all other hypotheses have been eliminated. The goal is to find policy Φ with minimum $\mathbb{E}[Cost(\Phi)]$. Again, the expectation is over the prior distribution of $\bar{i}$ as well as the outcomes of noisy tests.

Equivalently, we can view a policy as a *decision tree* with nodes corresponding to states, labels at nodes representing the test performed at that state and branches corresponding to the ±1 outcome at the current test. In particular, a non-adaptive policy is simply a decision tree where all nodes on each level are labelled with the same test.

As the number of states can be exponential, we cannot hope to specify arbitrary adaptive policies. Instead, we want implicit policies Φ, where given *any* state (E, d), the test $\Phi(E, d)$ can be computed *efficiently*. This would imply that the total time taken on any decision path is polynomial. We note that an optimal policy Φ^* can be very complex and the map $\Phi^*(E, d)$ may not be efficiently computable. We will still compare the performance of our (efficient) policy to Φ^*.

Noise Model. In this paper, we consider the *persistent noise* model. That is, repeating a test T with $\bar{i} \in T^*$ always produces the same outcome. An alternative model is non-persistent noise, where each run of test T with $\bar{i} \in T^*$ produces an independent random outcome. The persistent noise model is more appropriate to handle missing data. It also contains the non-persistent noise model as a special case (by introducing multiple tests with identical partitions). The persistent-noise model is also more challenging from an algorithmic point of view.

In fact, our results hold in a substantially more general setting (than ODT), that of covering arbitrary *submodular* functions. In Section II.4.2 we first describe this setting in the noiseless case, which is well-understood (prior to our work). Then, in Section II.4.3 we describe the setting with noisy outcomes, which is the focus of our paper.

II.4.2. Adaptive Submodular Ranking (Noiseless Case) We now review the (non-adaptive and adaptive) Submodular Ranking problems introduced by Azar and Gamzu (2011) and Navidi et al. (2020) respectively.

Submodular Function Ranking. An instance of Submodular Function Ranking (SFR) consists of a ground set of *elements* $[n] := \{1, ..., n\}$ and a collection of monotone submodular functions $\{f_1, ..., f_m\}$, $f_i : 2^{[n]} \to [0,1]$, with $f_i(\emptyset) = 0$ and $f_i([n]) = 1$ for all $i \in [m]$. Each $i \in [m]$ is called a *scenario*. An unknown *target* scenario $\bar{i}$ is drawn from a known distribution $\{\pi_i\}$ over $[m]$.

A solution to SFR is a permutation $\sigma = (\sigma(1), ..., \sigma(n))$ of elements. Given any such permutation, the *cover time* of scenario i is $C(i,\sigma) := \min\{t \mid f_i(\sigma^t) = 1\}$ where $\sigma^t = (\sigma(1), ..., \sigma(t))$ is the t-prefix of permutation σ. In words, the cover time is the earliest time when the value of f_i reaches the unit threshold. The goal is to find a permutation σ of $[n]$ with minimal expected cover time $\mathbb{E}_{\bar{i}}[C(\bar{i},\sigma)] = \sum_{i\in[m]} \pi_i \cdot C(i,\sigma)$.

The *separability* parameter $\varepsilon > 0$ is defined as minimum positive marginal increment of any function, i.e. $\varepsilon := \min\{f_i(S \cup \{e\}) - f_i(S) > 0 \mid \forall S \subseteq [n], i \in [m], e \in [n]\}$. We will use the following.

Theorem 1 **(Azar and Gamzu (2011))**. *There is an $O(\log \frac{1}{\epsilon})$-approximation algorithm for SFR.*

Adaptive Submodular Ranking. In the Adaptive Submodular Ranking (ASR) problem, in addition to the above input to SFR, for each scenario $i \in [m]$ we are given a *response function* $r_i : [n] \to \Omega$ where Ω is a finite set of *outcomes* (or response, which we use interchangeably). A solution to ASR is an *adaptive* sequence of elements: the sequence is adaptive because it can depend on the outcomes from previous elements. When the policy selects an element $e \in [n]$, it receives an outcome $o = r_{\bar{i}}(e) \in \Omega$, thereby any scenario i with $r_i(e) \neq \bar{o}$ can be ruled out.

The *state* of an adaptive policy is a tuple (E,d) where $E \subseteq [n]$ is the subset of previously selected elements and $d \in \Omega^E$ denotes the observed responses on E. An adaptive policy is then specified by a mapping $\Phi : 2^{[n]\times\Omega} \to [n]$ from states to elements, where $\Phi(E,d)$ is the next element to select at state (E,d). Note that any adaptive policy Φ induces, for each scenario i, a unique sequence σ_i of elements that will be selected if the target scenario $\bar{i} = i$. The *cover time* of i is defined as $C(i,\Phi) := \min\{t \mid f_i(\sigma_i^t) = 1\}$. The goal is to find a policy Φ with minimal expected cover time $\sum_{i\in[m]} \pi_i \cdot C(i,\Phi)$. We will use the following result in Section II.6.

Theorem 2 **(Navidi et al. (2020))**. *There is an $O(\log \frac{m}{\epsilon})$-approximation algorithm for ASR.*

As discussed in Navidi et al. (2020), the optimal decision tree problem (without noise) is a special case of ASR. We show later that even the noisy version ODTN can be reduced to a *noisy* variant of ASR (which we define next).

II.4.3. Adaptive Submodular Ranking with Noise In this paper, we introduce a new variant of ASR by incorporating noisy outcomes, which generalizes the ODTN problem.

ASR with Noise. An instance of the Adaptive Submodular Ranking with Noise (ASRN) Problem consists of a ground set of elements $[n]$, a finite set Ω of *outcomes*, and a collection of monotone submodular functions $\{f_1,...,f_m\}$, where each $f_i : 2^{[n]\times\Omega} \to [0,1]$ satisfies $f_i(\emptyset)=0$ and $f_i([n]\times\Omega)=1$. Note that the ground set of each function f_i is $[n]\times\Omega$, i.e., all element-outcome pairs. As before, each $i\in[m]$ is called a scenario and an unknown target scenario $\bar{i}$ is drawn from a given distribution $\{\pi_i\}_{i=1}^m$. For each scenario $i\in[m]$, we are given a *response function* $r_i : [n]\to\Omega\cup\{*\}$. When an element e is selected, its outcome is:

- $r_i(e)$ if $r_i(e)\in\Omega$, and
- a uniformly random response from Ω if $r_i(e)=*$ (noisy outcome).

The responses can be summarized in an $n\times m$ matrix M with entries from $\Omega\cup\{*\}$. Conditioned on $\bar{i}$, we assume that all noisy outcomes are independent. Our results extend to arbitrary distributions for noisy outcomes, but we will work with the uniform case for simplicity.

As in the noiseless case, the state of a policy is the tuple (E,d) where $E\subseteq[n]$ denotes the previously selected elements and $d\in\Omega^E$ denotes their observed responses. A *non-adaptive* policy is simply given by a permutation of all elements and involves selecting elements in this (static) sequence. An *adaptive* policy is a mapping $\Phi : 2^{[n]\times\Omega}\to[n]$, where $\Phi(E,d)$ is the next element to select at state (E,d). Scenario i is said to be *covered* in state (E,d) if $f_i(\{(e,d_e):e\in E\})=1$, i.e., function f_i is covered by the element-response pairs observed so far. The goal is to cover the target scenario $\bar{i}$ using the minimum expected number of elements.

Unlike the noiseless case, in ASRN, each scenario i may trace *multiple* paths in the decision tree corresponding to policy Φ. However, if we condition on the responses $\omega\in\Omega^n$ from all elements, each scenario i traces a unique path, corresponding to a sequence $\sigma_{i,\omega}$ of element-response pairs. The *cover time* of scenario i under ω is defined as $C(i,\Phi|\omega) := \min\{t|f_i(\sigma^t_{i,\omega})=1\}$ where $\sigma^t_{i,\omega}$ consists of the first t element-response pairs in $\sigma_{i,\omega}$. The expected cover time of scenario i is $\mathrm{ECT}(i,\Phi) := \sum_{\omega\in\Omega^n}\Pr(\omega|i)\cdot C(i,\Phi|\omega)$, where $\Pr(\omega|i)$ is the probability of observing responses ω conditioned on $\bar{i}=i$. Finally, the expected cost of policy Φ is $\sum_{i\in[m]}\pi_i\cdot\mathrm{ECT}(i,\Phi)$.

For each scenario i, we assume that the function f_i can always be covered irrespective of the noisy outcomes (when $\bar{i}=i$). In other words, for any $i\in[m]$ and $\omega\in\Omega^n$ that is *consistent* with scenario i (i.e., $\omega_e = r_i(e)$ for each e with $r_i(e) \neq *$), we must have $f_i(\{(e,\omega_e) : e \in [n]\}) = 1$. In the absence of this assumption, the optimal value (as defined above) will be unbounded.

Connection to ODTN. The ODTN problem can be cast as a special case of the ASRN problem, where the n tests $\mathcal{T}$ in ODTN corresponds to the elements $[n]$ in ASRN, and the m hypotheses in ODTN correspond to the scenarios in ASRN, with the same prior distribution. The outcomes $\Omega = \{\pm 1\}$. Define the response function for each test $T \in \mathcal{T}$ as follows. Let (T^+, T^-, T^*) be the 3-way partition of $[m]$ for test T. For any hypothesis (scenario) $i \in [m]$, define $r_i(T) = o$ if $i \in T^o$ for each $o \in \Omega \cup \{*\}$. For any $i \in [m]$, define the submodular function

$$f_i(S) = \frac{1}{m-1} \cdot | \bigcup_{T:(T,+1)\in S} T^- \bigcup \bigcup_{T:(T,-1)\in S} T^+ |, \quad \forall S \subseteq \mathcal{T} \times \{+1,-1\}.$$

Note that the element-outcome pairs here are $U = \mathcal{T} \times \{+1,-1\}$. It is easy to see that each function $f_i : 2^U \to [0,1]$ is monotone and submodular. Also, these functions f_i happen to be uniform for all i. Moreover, the separability parameter $\varepsilon = \frac{1}{m-1}$. Crucially, $f_i(S)$ corresponds to the fraction of hypotheses (other than i) that are incompatible with at least one outcome in S: for example, if S has a positive outcome $(T,+1)$ then hypotheses T^- are incompatible (similarly for negative outcomes). So f_i has value one exactly when i is identified as the only compatible hypothesis. By the assumption that the target hypothesis can be uniquely identified, the function f_i can be covered (i.e. reaches value one) irrespective of the noisy outcomes.

II.4.4. Expanded Scenario Set In our analysis for both the non-adaptive and adaptive ASRN problem, we will consider an equivalent noiseless ASR instance. Let $\mathcal{I}$ be a given ASRN instance with scenarios $[m]$. The ASR instance $\mathcal{J}$ considers an expanded set of scenarios. For any scenario $i \in [m]$, define

$$\Omega(i) := \{\omega \in \Omega^n : \omega_e = r_i(e) \text{ for all } e \in [n] \text{ with } r_i(e) \neq *\},$$

denoting all outcome vectors that are *consistent* with scenario i. For any $\omega \in \Omega(i)$, the *expanded* scenario (i,ω) corresponds to the original scenario $i \in [m]$ when the outcome of each element e is ω_e. Note that an expanded scenario also fixes all noisy outcomes. We write $H_i := \{(i,\omega) : \omega \in \Omega(i)\}$ and $H = \cup_{i=1}^m H_i$ for the set of all expanded scenarios.

To define the prior distribution in the ASR instance, let $c_i = |\{e \in [n] : r_i(e) = *\}|$ be the number of noisy outcomes for $i \in [m]$. Since the outcome of any $\star$-element for i is uniformly drawn from Ω, each of the $|\Omega|^{c_i}$ possible expanded scenarios for i occurs with the same probability $\pi_{i,\omega} = \pi_i/|\Omega|^{c_i}$.

To complete the reduction, for each $(i,\omega) \in H$, we define the response function

$$r_{i,\omega} : [n] \to \Omega, \quad r_{i,\omega}(e) = \omega_e, \qquad \forall e \in [n],$$

and the submodular coverage function

$$f_{i,\omega} : 2^{[n]} \to [0,1], \quad f_{i,\omega}(S) = f_i\big(\{(e,\omega_e) : e \in S\}\big), \qquad \forall S \subseteq [n].$$

By this definition, since f_i is monotone and submodular on $[n] \times \Omega$, the function $f_{i,\omega}$ is also monotone and submodular on $[n]$. We will also work with the ASR (noiseless) instance on the expanded scenarios with response functions $r_{i,\omega}$ and submodular functions $f_{i,\omega}$.

Proposition 1. *The ASRN instance $\mathcal{I}$ is equivalent to the ASR instance $\mathcal{J}$.*

Proof. Recall that an adaptive algorithm for ASRN or ASR can be viewed as a decision tree. We will show that any feasible decision tree for the ASR instance $\mathcal{J}$ is also feasible for the ASRN instance $\mathcal{I}$ with the same objective, and vice versa.

In one direction, consider a feasible decision tree $\mathbb{T}$ for the ASR instance $\mathcal{J}$. For any expanded scenario $(i,\omega) \in H$, let $P_{i,\omega}$ be the unique path traced in $\mathbb{T}$, and $S_{i,\omega}$ the elements selected along $P_{i,\omega}$. Note that by definition of a feasible decision tree, at the last node ("leaf") of path $P_{i,\omega}$, it holds $f_{i,\omega}(S_{i,\omega}) = 1$ which, in the notation of the original ASRN instance, translates to $f_i(\{(e,\omega_e) : e \in S_{i,\omega}\}) = 1$.

In the other direction, let $\mathbb{T}'$ be any decision tree for ASRN instance $\mathcal{I}$. Suppose the target scenario is $i \in [m]$ and element-outcomes are given by $\omega \in \Omega^n$ on the $\star$-elements for i, which is unknown to the algorithm. Then a *unique* path $P'_{i,\omega}$ is traced in $\mathbb{T}'$. Let $S'_{i,\omega}$ denote the elements on this path. Since i is covered at the end of $P'_{i,\omega}$ we have $f_i(\{(e,\omega_e) : e \in S'_{i,\omega}\}) = 1$. Now consider $\mathbb{T}'$ as a decision tree for ASR instance $\mathcal{J}$. Under scenario i,ω, it is clear that path $P'_{i,\omega}$ is traced and so elements $S'_{i,\omega}$ are selected. It follows that $f_{i,\omega}(S'_{i,\omega}) = f_i(\{(e,\omega_e) : e \in S'_{i,\omega}\}) = 1$ which means that scenario (i,ω) is covered at the end of $P'_{i,\omega}$. Therefore $\mathbb{T}'$ is a also feasible decision tree for $\mathcal{J}$. Taking expectations, the cost for $\mathcal{J}$ is at most that for instance $\mathcal{I}$. □

Crucially, the number of expanded scenarios $|H|$ is exponentially large as $|H| \le \sum_{i\in[m]} |\Omega|^{c_i}$. So we cannot merely apply existing algorithms for the noiseless ASR problems. In §II.5 and §II.6 we will show different ways for managing the expanded scenarios and obtaining *polynomial time* algorithms.

II.5. Nonadaptive Algorithm

This main result in this section is an $O(\log \frac{1}{\varepsilon})$-approximation for Non-Adaptive Submodular Function Ranking (SFRN) where $\varepsilon > 0$ is the separability parameter of the submodular functions. By Proposition 1, the SFRN problem is equivalent to the SFR problem on the expanded scenarios. However, as noted above, we cannot use Theorem 1 directly as the SFR instance has an exponential number of scenarios. Nevertheless, we can obtain the following result.

Theorem 3. *There is a poly$(\frac{1}{\varepsilon}, n, m)$ time $O(\log \frac{1}{\varepsilon})$-approximation for the SFRN problem.*

Observe that for ODTN, $\varepsilon = \frac{1}{m-1}$, thus we obtain the following result for ODTN.

Corollary 1. *There is an $O(\log m)$-approximation for non-adaptive ODTN.*

At a high level, our algorithm builds upon the algorithm of Azar and Gamzu (2011)'s greedy-style algorithm for SFR. In their algorithm, at any iteration, having already chosen elements E, assigns to each $e \in [n] \setminus E$ a score that measures the *coverage gain* when it is selected, defined as

$$G_E(e) := \sum_{(i,\omega)\in H: f_{i,\omega}(E)<1} \pi_{i,\omega} \frac{f_{i,\omega}(\{e\}\cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} = \sum_{(i,\omega)\in H} \pi_{i,\omega} \cdot \Delta_E(i,\omega;e), \tag{1}$$

$$\Delta_E(i,\omega,e) = \begin{cases} \frac{f_{i,\omega}(\{e\}\cup E) - f_{i,\omega}(E)}{1-f_{i,\omega}(E)}, & \text{if } f_{i,\omega}(E) < 1; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The algorithm then selects the element with the maximum score.

II.5.1. Non-adaptive Algorithm for SFRN

We now describe how to convert the above algorithm into a non-adaptive SFRN problem, formally described in Algorithm II.5.2. The algorithm involves two phases. In the first phase, we run the SFR algorithm using sampling to get estimates $\overline{G_E}(e)$ of the scores. If at some step, the maximum sampled score is "too low" then we go to the second phase where we perform all remaining elements in an arbitrary order. The number of samples used to obtain each estimate is polynomial in m, n, ε^{-1}, so the overall runtime is polynomial.

Pre-processing. We first show that by losing an $O(1)$-factor in approximation ratio, we may assume that $\pi_i \geq n^{-2}$ for all $i \in [m]$. Let $A = \{i \in [m] : \pi_i \leq n^{-2}\}$, then $\sum_i \pi_i \leq n^{-2} \cdot n \leq n^{-1}$. Replace all scenarios in A with a single dummy scenario "0" with $\pi_0 = \sum_{i\in A} \pi_i$, and define f_0 to be any f_i where $i \in A$. By our assumption that each f_i must be covered irrespective of the noisy outcomes, it holds that $f_{i,\omega}([n]) = 1$ for each $\omega \in \Omega(i)$, and hence the cover time is at most n. Thus, for any permutation σ, the expected cover time of the old and new instance differ by at most $O(n^{-1} \cdot n) = O(1)$. Therefore, the cover time of any sequence of elements differs by only $O(1)$

Algorithm 1 Non-adaptive SFRN algorithm.

```
Initialize $E \leftarrow \emptyset$ and sequence $\sigma = \emptyset$.
while $E \neq [n]$ do                                                          ▷ Phase 1 begins
    For each $e \in [n]$, compute an estimate $\overline{G_E}(e)$ of the score $G_E(e)$ by sampling from $H$ independently $N = m^3 n^4 \varepsilon^{-1}$ times.
    Let $e^*$ denote the element $e \in [n] \setminus E$ that maximizes $\overline{G_E}(e)$.
    if $\overline{G_E}(e) \geq \frac{1}{4} m^{-2} n^{-4} \varepsilon$ then
        Update $E \leftarrow E \cup \{e^*\}$ and append $e^*$ to sequence $\sigma$.
    else
        Exit the while loop.                                                   ▷ Phase 1 ends
Append the elements in $[n] \setminus E$ to sequence $\sigma$ in arbitrary order.   ▷ Phase 2
Output non-adaptive sequence $\sigma$.
```

in this new instance (where we removed the scenarios with tiny prior densities) and the original instances.

Since this summation involves exponentially many terms, we do not know how to compute the *exact* value of (1) in polynomial time. However, using the fact that $G_E(e)$ is the *expectation* of $\Delta_E(i, \omega; e)$ over the expanded scenarios $(i, \omega) \in H$, we will show how to obtain a randomized constant-approximate maximizer by sampling from H. Moreover, we use the following extension of Theorem 1, which follows directly from the analysis in Im et al. (2016).

THEOREM 4 **(Azar and Gamzu (2011), Im et al. (2016))**. *Consider the SFR algorithm that selects at each step, an element e with $G_E(e) \geq \Omega(1) \cdot \max_{e' \in U} G_E(e')$. This is an $O(\log \frac{1}{\epsilon})$-approximation algorithm.*

Consequently, if we always find an approximate maximizer for $G_E(e)$ by sampling then Theorem 3 would follow from Theorem 4. However, this sampling approach is not sufficient because it can fail when the value $G_E(e)$ is very small. In order to deal with this, a key observation is that when the score $G_E(e)$ is small *for all* elements e, then it must be that (with high probability) the already-selected elements E have covered $\bar{i}$, so any future elements would not affect the expected cover time. We describe how to overcome this challenge in the next subsection.

II.5.2. Analysis of Algorithm We now present the formal proof of Theorem 3. To analyze the our randomized algorithm, we need the following sampling lemma, which follows from the standard Chernoff bound.

Lemma 1. *Let X be a $[0,1]$-bounded random variable with $\mathbb{E}X \geq m^{-2}n^{-4}\varepsilon$. Let $\bar{X}$ denote the average of $m^3n^4\varepsilon^{-1}$ many independent samples of X. Then* $\Pr\left[\bar{X} \notin [\frac{1}{2}\mathbb{E}X, 2\mathbb{E}X]\right] \leq e^{-\Omega(m)}$.

Proof. Let $X_1, ..., X_N$ be i.i.d. samples of random variable where $N = m^3n^4\varepsilon^{-1}$ is the number of samples. Letting $Y = \sum_{i\in[N]} X_i$, the usual Chernoff bound implies for any $\delta \in (0,1)$,

$$\Pr\left(Y \notin [(1-\delta)\mathbb{E}Y, (1+\delta)\mathbb{E}Y]\right) \leq \exp(-\frac{\delta^2}{2}\cdot\mathbb{E}Y).$$

The lemma follows by setting $\delta = \frac{1}{2}$ and using the assumption $\mathbb{E}Y = N\cdot\mathbb{E}X_1 = \Omega(m)$. □

The next lemma shows that sampling does find an approximate maximizer unless the score is very small, and also bounds the *failure* probability.

Lemma 2. *Consider any step in the algorithm with $S = \max_{e\in[n]} G_E(e)$ and $\bar{S} = \max_{e\in[n]} \overline{G_E}(e)$ with $\overline{G_E}(e^*) = \bar{S}$. Call this step a failure if (i) $\bar{S} < \frac{1}{4}m^{-2}n^{-4}\varepsilon$ and $S \geq \frac{1}{2}m^{-2}n^{-4}\varepsilon$, or (ii) $\bar{S} \geq \frac{1}{4}m^{-2}n^{-4}\varepsilon$ and $G_E(e^*) < \frac{S}{4}$. Then the probability of failure is at most $e^{-\Omega(m)}$.*

Proof. We will consider the two types of failures separately. For the first type, suppose $S \geq \frac{1}{2}m^{-2}n^{-4}\varepsilon$. Using Lemma 1 on the element $e \in [n]$ with $G_E(e) = S$, we obtain

$$\Pr[\bar{S} < \frac{1}{4}m^{-2}n^{-4}\varepsilon] \leq \Pr[\overline{G_E}(e) < \frac{1}{4}m^{-2}n^{-4}\varepsilon] \leq e^{-\Omega(m)}.$$

So the probability of the first type of failure is at most $e^{-\Omega(m)}$.

For the second type of failure, we consider two further cases:

- $S < \frac{1}{8}m^{-2}n^{-4}\varepsilon$. For any $e \in [n]$ we have $G_E(e) \leq S < \frac{1}{8}m^{-2}n^{-4}\varepsilon$. Note that $\overline{G_E}(e)$ is the average of N independent samples each with mean $G_E(e)$. We now upper bound the probability of the event $\mathcal{B}_e$ that $\overline{G_E}(e) \geq \frac{1}{4}m^{-2}n^{-4}\varepsilon$. We first artificially increase each sample mean to $\frac{1}{8}m^{-2}n^{-4}\varepsilon$: note that this only increases the probability of event $\mathcal{B}_e$. Now, using Lemma 1 we obtain $\Pr[\mathcal{B}_e] \leq e^{-\Omega(m)}$. By a union bound, it follows that $\Pr[\bar{S} \geq \frac{1}{4}m^{-2}n^{-4}\varepsilon] \leq \sum_{e\in[n]} \Pr[\mathcal{B}_e] \leq e^{-\Omega(m)}$.
- $S \geq \frac{1}{8}m^{-2}n^{-4}\varepsilon$. Consider now any $e \in U$ with $G_E(e) < S/4$. By Lemma 1 (artificially increasing $G_E(e)$ to $S/4$ if needed), it follows that $\Pr[\overline{G_E}(e) > S/2] \leq e^{-\Omega(m)}$. Now consider the element e' with $G_E(e') = S$. Again, by Lemma 1, it follows that $\Pr[\overline{G_E}(e') \leq S/2] \leq e^{-\Omega(m)}$. This means that element e^* has $\overline{G_E}(e^*) \geq \overline{G_E}(e') > S/2$ and $G_E(e^*) \geq S/4$ with probability $1 - e^{-\Omega(m)}$. In other words, assuming $S \geq \frac{1}{8}m^{-2}n^{-4}\varepsilon$, the probability that $G_E(e^*) < S/4$ is at most $e^{-\Omega(m)}$.

Adding the probabilities over all possibilities for failures, the lemma follows. □

Based on Lemma 2, in the remaining analysis we condition on the event that our algorithm never encounters failures, which occurs with probability $1 - e^{-\Omega(m)}$. To conclude the proof, we need the following key lemma which essentially states that if the score of the greediest element is low, then the elements selected so far suffices to cover *all* scenarios with high probability, and hence the ordering of the remaining elements does not matter much.

LEMMA 3. *Assume that there are no failures. Consider the end of phase 1 in our algorithm, i.e. the first step with $\overline{G_E}(e^*) < \frac{1}{4}m^{-2}n^{-4}\varepsilon$. Then, the probability that the realized scenario is not covered is at most m^{-2}.*

Proof. Let E denote the elements chosen so far and p the probability that E does *not* cover the realized scenario-copy of H. That is,

$$p = \Pr_{(i,\omega)\in H}(f_{i,\omega}(E) < 1) = \sum_{i=1}^{m} \pi_i \cdot \Pr_{\omega\in\Omega(i)}(f_{i,\omega}(E) < 1).$$

It follows that there is some i with $\Pr_{\omega\in\Omega(i)}(f_{i,\omega}(E) < 1) \geq p$. By definition of separability, if $f_{i,\omega}(E) < 1$ then $f_{i,\omega}(E) \leq 1-\varepsilon$. Thus,

$$\sum_{\omega\in\Omega(i)} \pi_{i,\omega} f_{i,\omega}(E) \leq \sum_{\omega: f_{i,\omega}(E)=1} \pi_{i,\omega}\cdot 1 + \sum_{\omega: f_{i,\omega}(E)<1} \pi_{i,\omega}\cdot f_{i,\omega}(E) \leq (1-\varepsilon p)\pi_i.$$

On the other hand, taking all the elements we have $f_{i,\omega}([n]) = 1$ for all $\omega \in \Omega(i)$. Thus,

$$\sum_{\omega\in\Omega(i)} \pi_{i,\omega} f_{i,\omega}([n]) = \sum_{\omega\in\Omega(i)} \pi_{i,\omega} = \pi_i.$$

Taking the difference of the above two inequalities, we have

$$\sum_{\omega\in\Omega(i)} \pi_{i,\omega}\cdot (f_{i,\omega}([n]) - f_{i,\omega}(E)) \geq \pi_i \cdot \varepsilon p.$$

Consider function $g(S) := \sum_{\omega\in\Omega(i)} \pi_{i,\omega}\cdot (f_{i,\omega}(S\cup E) - f_{i,\omega}(E))$ for $S \subseteq [n]$, which is also submodular. From the above, we have $g([n]) \geq \pi_i \cdot \varepsilon p$. Using submodularity of g,

$$\max_{e\in[n]} g(\{e\}) \geq \frac{\varepsilon p \pi_i}{n} \implies \exists \tilde{e} \in [n] : \sum_{\omega\in\Omega(i)} \pi_{i,\omega}\cdot (f_{i,\omega}(E\cup\{\tilde{e}\}) - f_{i,\omega}(E)) \geq \frac{\varepsilon p \pi_i}{n}.$$

It follows that $G_E(\tilde{e}) \geq \frac{\varepsilon p \pi_i}{n} \geq n^{-3}\varepsilon p$, where we used that $\min_i \pi_i \geq n^{-2}$. Now, suppose for a contradiction that $p \geq m^{-2}$. Since there is no failure and $G_E(\tilde{e}) \geq n^{-3}m^{-2}\varepsilon \geq \frac{1}{4}n^{-4}m^{-2}\varepsilon$, by case (ii) of Lemma 2 , we deduce that $\overline{G_E}(e^*) \geq \frac{1}{4}m^{-2}n^{-4}$, which is contradiction. □

The above is essentially a consequence of the submodularity of the target functions. Suppose for contradiction that there is a scenario i that, with at least m^{-2} probability over the random outcomes, remains *uncovered* by the currently selected elements. Recall that by our feasibility assumption, if all elements were selected, then f_i is covered with probability 1. Thus, by submodularity, there exists an individual element $\tilde{e}$ whose inclusion brings more coverage than the average coverage over all elements in $[n]$, and hence $\tilde{e}$ has a "high" score.

Proof of Theorem 3. Assume that there are no failures. We proceed by bounding the expected costs (number of elements) from phase 1 and 2 separately. By Lemma 2, the element chosen in each step of phase 1 is a 4-approximate maximizer (see case (ii) failure) of the score used in the SFR algorithm. Thus, by Theorem 4, the expected cost in phase 1 is $O(\log m)$ times the optimum. On the other side, by Lemma 3 the probability of performing phase 2 is at most $e^{-\Omega(m)}$. As there are at most n elements in phase 2, the expected cost is only $O(1)$. Therefore, Algorithm II.5.2 is an $O(\log m)$-approximation algorithm for SFRN. $\square$

II.6. Adaptive Algorithms

In this section we present the $O\left(\log \frac{m}{\varepsilon} + \min\{c \log |\Omega|, r\}\right)$-approximation for ASRN where we recall that c, r are the maximum number of noisy entries ("stars") per column and per row in the outcome matrix M, and ε is the separability parameter of the submodular functions. We propose two algorithms, achieving $O\left(r + \log \frac{m}{\varepsilon}\right)$ and $O\left(c \log |\Omega| + \log \frac{m}{\varepsilon}\right)$ approximations respectively, which combined imply our main result.

In both algorithms, we maintain the posterior probability of each scenario based on the previous element responses, and use these probabilities to calculate a *score* for each element, which comprises (i) a term that prioritizes splitting the candidate scenarios in a balanced manner and (ii) terms corresponding to the expected number of scenarios eliminated. Different than the noiseless setting, in ASRN (and ODTN), each scenario may trace *multiple* paths in the decision tree due to outcome randomness. In fact, each scenario may trace an exponential number of paths in the tree, so a naive generalization of the analysis in Navidi et al. (2020) incurs an extra exponential factor in the approximation ratio.

We circumvent this challenge by reducing to an ASR instance $\mathcal{J}$ (as defined in Proposition 1) using the *expanded* scenarios. In this way, the noise is removed, since we recall that the outcome of each element is deterministic *conditional* on any expanded scenario (i, ω). Our first result, an $O(c \log |\Omega| + \log \frac{m}{\varepsilon})$-approximation, then follows from Navidi et al. (2020).

However, as $\mathcal{J}$ involves exponentially many scenarios, a naive implementation of the algorithm in Navidi et al. (2020) leads to exponential running time. To improve the computational efficiency, in Section II.6.1 we exploit the special structure of $\mathcal{J}$ and devise a polynomial time algorithm. Then, in Section II.6.2, we propose a slightly different algorithm than that of Navidi et al. (2020), and show an $O(r + \log \frac{m}{\varepsilon})$ approximation ratio.

II.6.1. An $O(c\log|\Omega| + \log\frac{m}{\varepsilon})$-Approximation Algorithm Our first adaptive algorithm is based on the $O(\log\frac{m}{\varepsilon})$-approximation algorithm for ASR from Navidi et al. (2020), formally stated as Algorithm 2. Applying this result on the instance $\mathscr{J}$ and recalling $|H| \le |\Omega|^c \cdot m$, we immediately obtain the desired guarantee. Their algorithm, rephrased in our notations, maintains the set $H' \subseteq H$ of all expanded scenarios that are *consistent* with all the observed outcomes, and iteratively selects the element with maximum score, as defined in (3)[‡].

As the heart of the algorithm, this score strikes a balance between *covering* the submodular functions of the consistent scenarios and *shrinking* H' hence reducing the uncertainty in the target scenario. The second term in Score_c, similar to the score in our non-adaptive algorithm (Algorithm II.5.2), involves the sum of the incremental coverage (for selecting e) over all uncovered expanded scenarios, weighted by their current coverage, with higher weights on the expanded scenarios closer to being covered.

To interpret the first term in Score_c, let us for simplicity assume $\Omega = \{\pm 1\}$ and $\pi_{i,\omega}$ is uniform over H. Upon selecting an element, H' is split into two subsets, among which $L_e(H')$ is the lighter (in cardinality), or equivalently – since we just assumed $\pi_{i,\omega}$ to be uniform – in the total prior probabilities. Thus, this term is simply the number of expanded scenarios eliminated in the worst case (over the outcomes in Ω). This is reminiscent of the greedy algorithm for the ODT problem (e.g. Kosaraju et al. (1999)) which iteratively selects a test that maximizes the number of scenarios ruled out, in the *worst* case over all test outcomes. Evidently, the higher this term, the more progress is made towards identifying the target (expanded) scenario.

As noted earlier, a key issue is the exponential size of the expanded scenario set H. The naive implementation, which computes the summation in Score_c by evaluating each term in H', requires exponential time. Nonetheless, as the main focus of this subsection, we explain how to utilize the structure of the ASRN instance $\mathscr{J}$ to reformulate each of the two terms in Score_c in a manageable form, hence enabling a polynomial time implementation.

Computing the First Term in Score_c. Recall that H_i is the set of all expanded scenarios for i. Since each $(i,\omega) \in H_i$ is has an equal share $\pi_{i,\omega} = |\Omega|^{-c_i}\pi_i$ of prior probability mass the (original) scenario $i \in [m]$, computing the first term in $Score_c$ reduces to maintaining the *number* $n_i = |H_i \cap H'|$ of consistent copies of i. We observe that n_i can be easily updated in each iteration. In fact, suppose outcome $o \in \Omega$ is observed upon selecting element e. We consider how $H' \cap H_i$ changes after selecting in the following three cases.

[‡] We use the subscript c to distinguish from the score function Score_r considered in Section II.6.2, but for ease of notation, we will suppress the subscript in this subsection.

Algorithm 2 Algorithm for ASR instance $\mathcal{J}$, based on Navidi et al. (2020).

```
Initialize $E \leftarrow \emptyset, H' \leftarrow H$.
while $H' \neq \emptyset$ do
    For any element $e \in [n]$, let $B_e(H')$ be the largest cardinality set among
        $\{(i,\omega) \in H' : r_{i,\omega}(e) = o\} \qquad \forall o \in \Omega$
    Define $L_e(H') = H' \setminus B_e(H')$
    Select the element $e \in [n] \setminus E$ maximizing
        $\text{Score}_c(e, E, H') = \pi\big(L_e(H')\big) + \sum_{(i,\omega) \in H', f_{i,\omega}(E) < 1} \pi_{i,\omega} \cdot \frac{f_{i,\omega}(e \cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} \qquad (3)$
    Observe response $o$ and update $H'$ as $H' \leftarrow \{(i,\omega) \in H' : \omega_e = o \text{ and } f_{i,\omega}(E \cup e) < 1\}$
    $E \leftarrow E \cup \{e\}$
```

1. if $r_i(e) \notin \{\star, o\}$, then none of i's expanded scenarios would remain in H', so n_i becomes 0,
2. if $r_i(e) = o$, then all of i's expanded scenarios would remain in H', so n_i remains the same,
3. if $r_i(e) = \star$, then only those (i,ω) with $\omega(e) = o$ will remain, and so n_i shrinks by an $|\Omega|$ factor.

As n_i's can be easily updated, we are also able to compute the first term in Score_c efficiently. Indeed, for any element e (that is not yet selected), we can implicitly describe the set $L_e(H')$ as follows. Note that for any outcome $o \in \Omega$,

$$|\{(i,\omega) \in H' : r_{i,\omega}(e) = o\}| = \sum_{i \in [m] : r_i(e) = o} n_i + \frac{1}{|\Omega|} \sum_{i \in [m] : r_i(e) = \star} n_i,$$

so the largest cardinality set $B_e(H')$ can then be easily determined using n_i's. In fact, let b be the outcome corresponding to $B_e(H')$. Then,

$$\pi\left(L_e\left(H'\right)\right) = \sum_{i \in [m] : r_i(e) \notin \{b, \star\}} \frac{\pi_i}{|\Omega|^{c_i}} \cdot n_i + \frac{|\Omega| - 1}{|\Omega|} \sum_{i \in [m] : r_i(e) = \star} \frac{\pi_i}{|\Omega|^{c_i}} \cdot n_i.$$

Computing the Second Term in Score_c**.** The second term in Score_c involves summing over exponentially many terms, so a naive implementation is inefficient. Instead, we will rewrite this summation as an *expectation* that can be calculated in polynomial time.

We introduce some notations before formally stating this equivalence. Suppose the algorithm selected a subset E of elements, and observed outcomes $\{\nu_e\}_{e \in E}$. We overload notation slightly and use $f(\nu_E) := f\big(\{(e, \nu_e) : e \in E\}\big)$ for any function f defined on $2^{[m] \times \Omega}$. For each scenario $i \in [m]$, let $p_i = n_i \cdot \frac{\pi_i}{|\Omega|^{c_i}}$ be the total probability mass of the surviving expanded scenarios for i.[†] Finally, for any element e and scenario i, let $\mathbb{E}_{i,\nu_e}$ be the expectation over the outcome ν_e of element e conditional on i being the realized scenario. We can then rewrite the second term in Score_c as follows.

Lemma 4. *For each $i \in [m]$, and $e \notin E$,*

$$\sum_{(i,\omega)\in H'} \pi_{i,\omega} \cdot \frac{f_{i,\omega}(e \cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} = \sum_{i\in[m]} p_i \cdot \frac{\mathbb{E}_{i,\nu_e}[f_i(\nu_E \cup \{\nu_e\}) - f_i(\nu_E)]}{1 - f_i(\nu_E)} \tag{4}$$

Proof. By decomposing the summation in the left hand side of (3) as $H' = \cup_i H' \cap H_i$, and noticing that $f_{i,\omega}(E) = f_i(\nu_E)$, the problem reduces to showing that for each $i \in [m]$,

$$\sum_{(i,\omega)\in H'\cap H_i} \pi_{i,\omega} \cdot \big(f_{i,\omega}(e \cup E) - f_{i,\omega}(E)\big) = p_i \cdot \mathbb{E}_{i,\nu_e}[f_i(\nu_E \cup \{\nu_e\}) - f_i(\nu_E)].$$

Recall that $p_i = n_i \cdot \frac{\pi_i}{|\Omega|^{c_i}}$ and $\pi_{(i,\omega)} = \frac{\pi_i}{|\Omega|^{c_i}}$, the above simplifies to

$$\frac{1}{n_i} \sum_{(i,\omega)\in H'\cap H_i} \big(f_{i,\omega}(e \cup E) - f_{i,\omega}(E)\big) = \mathbb{E}_{i,\nu_e}[f_i(\nu_E \cup \{\nu_e\}) - f_i(\nu_E)].$$

Note that $n_i = |H' \cap H_i|$, so the above is equivalent to

$$\frac{1}{n_i} \sum_{(i,\omega)\in H'\cap H_i} f_{i,\omega}(e \cup E) = \mathbb{E}_{i,\nu_e}[f_i\big(\nu_E \cup \{\nu_e\}\big)]. \tag{5}$$

It is straightforward to verify that the above by considering the following are two cases.

- If $r_i(e) = \nu_e \in \Omega \setminus \{*\}$, then the outcome ν_e is deterministic conditional on scenario i, and so is $f_i\big(\nu_E \cup \{\nu_e\}\big)$, the value of f_i after selecting e. On the left hand side, for every $\omega \in H_i$, by definition of H_i it holds $\nu_e = \omega_e$, and hence $f_{i,\omega}(e \cup E) = f_i(\nu_E \cup \{\nu_e\}$ for *every* $(i,\omega) \in H_i$. Therefore all terms in the summation are equal to $f_i(\nu_E \cup \{\nu_e\}$ and hence (5) holds.
- If $r_i(e) = *$, then each outcome $o \in \Omega$ occurs with equal probabilities, thus we may rewrite the right hand side as

$$\begin{aligned}\mathbb{E}_{i,\nu_e}[f_i\big(\nu_E \cup \{\nu_e\}\big)] &= \sum_{o\in\Omega} \mathbb{P}_i[\nu_e = o] \cdot f_i\big(\nu_E \cup \{\nu_e\}\big) \\ &= \frac{1}{|\Omega|} \sum_{o\in\Omega} f_i\big(\nu_E \cup \{(e,o)\}\big).\end{aligned}$$

[†] One may easily verify via the Bayesian rule that $p_i/p([m])$ is indeed the posterior probability of scenario $i \in [m]$, given the previously observed outcomes.

To analyze the other side, note that by definition of H_i and H', there are equally many expanded scenarios (i,ω) in $H' \cap H_i$ with $\omega_e = o$ for each outcome $o \in \Omega$. Thus, we can rewrite the left hand side as

$$\begin{aligned}\frac{1}{n_i} \sum_{(i,\omega)\in H'\cap H_i} f_{i,\omega}(e \cup E) &= \frac{1}{n_i} \sum_{o\in\Omega} \sum_{\substack{(i,\omega)\in H'\cap H_i,\\ \omega_e = o}} f_{i,\omega}(e \cup E) \\ &= \frac{1}{n_i} \sum_{o\in\Omega} \frac{n_i}{|\Omega|} f_{i,\omega}(e \cup E) \\ &= \frac{1}{|\Omega|} \sum_{o\in\Omega} f_i\big(\nu_E \cup \{(e,o)\}\big),\end{aligned}$$

which matches the right hand side of (5) and completes the proof. □

The above lemma suggests the following efficient implementation of Algorithm 2. For each i, compute and maintain p_i using n_i. To find the expectation in the numerator, note that if $r_i(e) \neq \star$, then ν_e is deterministic and hence it is straightforward to find this expectation. In the other case, if $r_i(e) = \star$, recalling that the outcome is uniform over Ω, we may simply evaluate $f_i(\nu_E \cup \{(e,o)\}) - f_i(\nu_E)$ for each $o \in \Omega$ and take the average, since the noisy outcome is uniformly distributed over Ω.

Now we are ready to formally state and prove the main result of this subsection.

Theorem 5. *Algorithm 2 is an $O(c\log|\Omega| + \log m + \log\frac{1}{\varepsilon})$-approximation algorithm for ASRN where c is the maximum number of noisy outcomes in each column of the response matrix M.*

Proof. Consider the ASR instance $\mathcal{J}$ and Algorithm 2. As discussed above, this algorithm can be implemented in polynomial time. By Theorem 2, this algorithm has an $O\big(\log(|\Omega|^c m) + \log\frac{m}{\varepsilon}\big) = O(c\log|\Omega| + \log\frac{m}{\varepsilon})$ approximation ratio since $|H| \le |\Omega|^c \cdot m$. □

II.6.2. An $O(r + \log\frac{m}{\varepsilon})$-Approximation Algorithm In this section, we consider a slightly different score function, Score_r, and obtain an $O(r + \log\frac{m}{\varepsilon})$-approximation. Unlike the previous section where the approximation factor follows as an immediately corollary from Theorem 2, to prove this result, we need to also modify the analysis.

The only difference from Algorithm 2 is in the first term of the score function. Recall that in Score_c, upon selecting an element, the surviving expanded scenarios is partitioned into $|\Omega|$ subsets, among which $L_e(H')$ is defined to be the lightest cardinality. Its counterpart in Score_r, however, is defined more indirectly, by first considering the *original* scenarios. The element e partitions the original scenarios with *deterministic* outcomes into $|\Omega|$ subsets, with the largest (in cardinality)

Algorithm 3 Modified algorithm for ASR instance $\mathcal{J}$.

```
Initialize $E \leftarrow \emptyset, H' \leftarrow H$
while $H' \neq \emptyset$ do
    $S \leftarrow \{i \in [m] : H_i \cap H' \neq \emptyset\}$    ▷ Consistent original scenarios
    For $e \in [n]$, let $U_e(S) = \{i \in S : r_i(e) = *\}$ and $C_e(S)$ be the largest cardinality set among
        $\{i \in S : r_i(e) = o\}, \quad \forall o \in \Omega,$
    and let $o_e(S) \in \Omega$ be the outcome corresponding to $C_e(S)$.
    For each $e \in [n]$, let
        $\overline{R_e}(H') = \{(i,\omega) \in H' : i \in C_e(S)\} \bigcup \{(j, o_e(S)) \in H' : j \in U_e(S)\},$
    be those expanded-scenarios that have outcome $o_e(S)$ for element $e$, and $R_e(H') := H' \setminus \overline{R_e}(H')$.
    Select element $e \in [n] \setminus E$ that maximizes
        $\mathrm{Score}_r(e, E, H') = \pi\big(R_e(H')\big) + \sum_{(i,\omega) \in H', f_{i,\omega}(E) < 1} \pi_{i,\omega} \cdot \frac{f_{i,\omega}(e \cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)}$    (6)
    Observe outcome $o$
    $H' \leftarrow \{(i,\omega) \in H' : r_{i,\omega}(e) = o \text{ and } f_{i,\omega}(E \cup e) < 1\}$    ▷ Update the (expanded) scenarios
    $E \leftarrow E \cup \{e\}$
```

being $C_e(S) \subseteq [m]$. The set $R_e(H') \subseteq H'$ is then defined to be the consistent expanded scenarios that have a different outcome than $C_e(S)$.

Computational Complexity. By definition, S can be directly computed using the n_i's, which can be updated in polynomial time as explained in Section II.6.1. Similar to Algorithm 2, the second term here also involves summing over exponentially many terms, but by following the same recipe as in Section II.6.1, one may also implement it in polynomial time.

The main result of this section, stated below, is proved by adapting the proof technique from Navidi et al. (2020).

THEOREM 6. *Algorithm 3 is a polynomial-time $O(r + \log \frac{m}{\varepsilon})$-approximation algorithm for ASRN, where r is the maximum number of noisy outcomes in any row of the response matrix M.*

Proof. The proof is similar to the analysis in Navidi et al. (2020). With some foresight, set $\alpha := 15(r + \log m)$. Write Algorithm 3 as ALG and let OPT be the optimal adaptive policy. It will be convenient to view ALG and OPT as decision trees where each node represents the "state" of the policy. Nodes in the decision tree are labelled by elements (that are selected at the corresponding state) and branches out of each node are labelled by the outcome observed at that point. At any state, we use E to denote the previously selected elements and $H' \subseteq M$ to denote the *expanded-scenarios* that are (i) compatible with the outcomes observed so far and (ii) uncovered. Suppose at some iteration, elements E are selected and outcomes ν_E are observed, then a scenario i is said to be *covered* if $f_i(E \cup \nu_E) = 1$, and *uncovered* otherwise.

For ease of presentation, we use the phrase "at time t" to mean "after selecting t elements". Note that the cost incurred until time t is exactly t. The key step is to show

$$a_k \leq 0.2 a_{k-1} + 3 y_k, \qquad \text{for all } k \geq 1, \tag{7}$$

where

- $A_k \subseteq M$ is the set of uncovered expanded scenarios in ALG at time $\alpha \cdot 2^k$ and $a_k = p(A_k)$ is their total probability,
- Y_k is the set of uncovered scenarios in OPT at time 2^{k-1}, and $y_k = p(Y_k)$ is the total probability of these scenarios.

As shown in Section 2 of Navidi et al. (2020), (7) implies that Algorithm 3 is an $O(\alpha)$-approximation and hence Theorem 6 follows. To prove (7), we consider the total score collected by ALG between iterations $\alpha 2^{k-1}$ and $\alpha 2^k$, formally given by

$$Z := \sum_{t > \alpha 2^{k-1}}^{\alpha 2^k} \sum_{(E,H') \in V(t)} \max_{e \in [n] \setminus E} \left(\sum_{(i,\omega) \in R_e(H')} \pi_{i,\omega} + \sum_{(i,\omega) \in H'} \pi_{i,\omega} \cdot \frac{f_{i,\omega}(e \cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} \right) \tag{8}$$

where $V(t)$ denotes the set of states (E, H') that occur at time t in the decision tree ALG. We note that all the expanded-scenarios seen in states of $V(t)$ are contained in A_{k-1}.

Consider any state (E, H') at time t in the algorithm. Recall that H' are the expanded-scenarios and let $S \subseteq [m]$ denote the original scenarios in H'. Let $T_{H'}(k)$ denote the subtree of OPT that corresponds to paths traced by expanded-scenarios in H' up to time 2^{k-1}. Note that each node (labeled by any element $e \in [n]$) in $T_H(k)$ has at most $|\Omega|$ outgoing branches and one of them corresponds to the outcome $o_e(S)$ defined in Algorithm 3. We define $\mathsf{Stem}_k(H')$ to be the path in $T_{H'}(k)$ that at each node (labeled e) follows the $o_e(S)$ branch. We also use $\mathsf{Stem}_k(H') \subseteq [n] \times \Omega$ to denote the observed element-outcome pairs on this path.

Definition 1. Each state (E, H') is exactly one of the following types:

- **bad** if the probability of uncovered scenarios in H' at the end of $\mathsf{Stem}_k(H')$ is at least $\frac{\Pr(H')}{3}$.
- **okay** if it is not bad and $\Pr(\cup_{e \in \mathsf{Stem}_k(H')} R_e(H'))$ is at least $\frac{\Pr(H')}{3}$.

- **good** if it is neither bad nor okay and the probability of scenarios in H' that get covered by $\mathsf{Stem}_k(H')$ is at least $\frac{\Pr(H')}{3}$.

Crucially, this categorization of states is well defined. Indeed, each expanded-scenario in H' is **(i)** uncovered at the end of $\mathsf{Stem}_k(H')$, or **(ii)** in $R_e(H')$ for some $e \in \mathsf{Stem}_k(H')$, or **(iii)** covered by some prefix of $\mathsf{Stem}_k(H')$, i.e. the function value reaches 1 on $\mathsf{Stem}_k(H')$. So the total probability of the scenarios in one of these 3 categories must be at least $\frac{\Pr(H)}{3}$.

In the next two lemmas, we will show a lower bound (Lemma 5) and an upper bound (Lemma 6) for Z in terms of a_k and y_k, which together imply (7) and complete the proof.

Lemma 5. *For any $k \geq 1$, it holds $Z \geq \alpha \cdot (a_k - 3y_k)/3$.*

Proof. The proof of this lower bound is identical to that of Lemma 3 in Navidi et al. (2020) for noiseless-ASR. The only difference is that we use the scenario-subset $R_e(H') \subseteq H'$ instead of subset "$L_e(H) \subseteq H$" in the analysis of Navidi et al. (2020). □

Lemma 6. *For any $k \geq 1$, $Z \leq a_{k-1} \cdot (1 + \ln \frac{1}{\epsilon} + r + \log m)$.*

Proof. This proof is analogous to that of Lemma 4 in Navidi et al. (2020) but requires new ideas, as detailed below. Our proof splits into two steps. We first rewrite Z by interchanging its double summation: the outer layer is now over the A_{k-1} (instead of times between $\alpha 2^{k-1}$ to $\alpha 2^k$ as in the original definition of Z). Then for each fixed $(i,\omega) \in A_{k-1}$, we will upper bound the inner summation using the assumption that there are at most r original scenarios with $r_i(e) = \star$ for each element e.

Step 1: Rewriting Z. For any uncovered $(i,\omega) \in A_{k-1}$ in the decision tree ALG at time $\alpha 2^{k-1}$, let $P_{i,\omega}$ be the path traced by (i,ω) in ALG, starting from time $\alpha 2^{k-1}$ and ending at time $\alpha 2^k$ or when (i,ω) is covered.

Recall that in the definition of Z, for each time t between $\alpha 2^{k-1}$ and $\alpha 2^k$, we sum over all states (E,H') at time t. Since $t \geq \alpha 2^{k-1}$, and the subset of uncovered scenarios only shrinks at t increases, for any $(E,H') \in V(t)$ we have $H' \subseteq A_{k-1}$. So, only the expanded scenarios in A_{k-1} contribute to Z. Thus we may rewrite (8) as

$$\begin{aligned} Z &= \sum_{(i,\omega)\in A_{k-1}} \pi_{i,\omega} \cdot \sum_{(e;E,H')\in P_{i,\omega}} \left(\frac{f_{i,\omega}(e\cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} + \mathbf{1}[(i,\omega)\in R_e(H')] \right) \\ &\leq \sum_{(i,\omega)\in A_{k-1}} \pi_{i,\omega} \cdot \left(\sum_{(e;E,H')\in P_{i,\omega}} \frac{f_{i,\omega}(e\cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} + \sum_{(e;E,H')\in P_{i,\omega}} \mathbf{1}[(i,\omega)\in R_e(H')] \right). \end{aligned} \tag{9}$$

Step 2: Bounding the Inner Summation. The rest of our proof involves upper bounding each of the two terms in the summation over $e \in P_{i,\omega}$ for any fixed $(i,\omega) \in A_{k-1}$. To bound the first term, we need the following standard result on submodular functions.

Lemma 7 **(Azar and Gamzu (2011)).** *Let $f : 2^U \to [0,1]$ be any monotone function with $f(\emptyset) = 0$ and $\varepsilon = \min\{f(S \cup \{e\}) - f(S) : e \in U, S \subseteq U, f(S \cup \{e\}) - f(S) > 0\}$ be the separability parameter. Then for any nested sequence of subsets $\emptyset = S_0 \subseteq S_1 \subseteq \cdots S_k \subseteq U$, it holds*

$$\sum_{t=1}^{k} \frac{f(S_t) - f(S_{t-1})}{1 - f(S_{t-1})} \leq 1 + \ln \frac{1}{\varepsilon}.$$

It follows immediately that

$$\sum_{(e;E,H') \in P_{i,\omega}} \frac{f_{i,\omega}(e \cup E) - f_{i,\omega}(E)}{1 - f_{i,\omega}(E)} \leq 1 + \ln \frac{1}{\varepsilon}. \tag{10}$$

Next we consider the second term $\sum_{(e;E,H') \in P_{i,\omega}} \mathbf{1}[(i,\omega) \in R_e(H')]$. Recall that $S \subseteq [m]$ is the subset of original scenarios with at least one expanded scenario in H'. Consider the partition of scenarios S into $|\Omega| + 1$ parts based on the response entries (from $\Omega \cup \{*\}$) for element e. From Algorithm 3, recall that $U_e(S)$ denotes the part with response $*$ and $C_e(S)$ denotes the largest cardinality part among the non-$*$ responses. Also, $o_e(S) \in \Omega$ is the outcome corresponding to part $C_e(S)$. Moreover, $R_e(H') \subseteq H'$ consists of all expanded-scenarios that *do not* have outcome $o_e(S)$ on element e. Suppose that $(i,\omega) \in R_e(H')$. Then, it must be that the observed outcome on e is *not* $o_e(S)$. Let $S' \subseteq S$ denote the subset of original scenarios that are also compatible with the observed outcome on e. We now claim that $|S'| \leq \frac{|S|+r}{2}$. To see this, let $D_e(S) \subseteq S$ denote the part having the *second largest cardinality* among the non-$*$ responses for e. As the observed outcome is not $o_e(S)$ (which corresponds to the largest part), we have

$$|S'| \leq |U_e(S)| + |D_e(S)| \leq |U_e(S)| + \left(\frac{|S| - |U_e(S)|}{2}\right) = \frac{|S| + |U_e(S)|}{2} \leq \frac{|S| + r}{2}.$$

The first inequality above uses the fact that S' consists of $U_e(S)$ (scenarios with $*$ response) and some part (other than $C_e(S)$) with a non-$*$ response. The second inequality uses $|D_e(S)| \leq \frac{|D_e(S)|+|C_e(S)|}{2} \leq \frac{|S|-|U_e(S)|}{2}$. The last inequality uses the upper-bound r on the number of $*$ responses per element. It follows that each time $(i,\omega) \in R_e(H')$, the number of compatible (original) scenarios on path $P_{i,\omega}$ changes as $|S'| \leq \frac{|S|+r}{2}$. Hence, after $\log_2 m$ such events, the number of compatible scenarios on path $P_{i,\omega}$ is at most r. Finally, we use the fact that the number of compatible scenarios reduces by at least one whenever $(i,\omega) \in R_e(H')$, to obtain

$$\sum_{(e;E,H') \in P_{i,\omega}} \mathbf{1}[(i,\omega) \in R_e(H')] \leq r + \log_2 m. \tag{11}$$

Combining (9), (10) and (11), we obtain the lemma. □

Combining the above result with Theorem 5 and selecting the one with lower approximation ratio between Algorithm 2 and Algorithm 3, we immediately obtain the following.

Theorem 7. *There is an adaptive $O\big(\min\{c\log|\Omega|, r\} + \log\frac{m}{\varepsilon}\big)$-approximation algorithm for the ASRN problem.*

II.6.3. Application of Algorithm 2 and Algorithm 3 to ODTN. When applied to the ODTN problem, Theorem 7 implies an $O\big(\min\{c, r\} + \log\frac{m}{\varepsilon}\big)$-approximation algorithm, which is also used in our computational results.

For concreteness, we provide a closed-form formula for Score_c and Score_r in the ODTN problem using Lemma 4, which were used in our experiments for ODTN. In §II.4.3, we formulated ODTN as an ASRN instance. Recall that the outcomes $\Omega = \{+1, -1\}$, and the submodular function f (associated with each hypothesis i) measures the proportion of hypotheses eliminated after observing the outcomes of a subset of tests.

As in §II.6, at any point in Algorithm 2 or 3, after selecting set E of tests, let $\nu_E : E \to \pm 1$ denote their outcomes. For each hypothesis $i \in [m]$, let n_i denote the number of surviving expanded-scenarios of i. Also, for each hypothesis i, let p_i denote the total probability mass of the surviving expanded-scenarios of i. For any $S \subseteq [m]$, we use the shorthand $p(S) = \sum_{i\in S} p_i$. Finally, let $A \subseteq [m]$ denote the compatible hypotheses based on the observed outcomes ν_E (these are all the hypotheses i with $n_i > 0$). Then, $f(\nu_E) = \frac{m-|A|}{m-1}$. Moreover, for any new test/element T,

$$f(\nu_E \cup \{\nu_T\}) = \begin{cases} \frac{m-|A|+|A\cap T^-|}{m-1} & \text{if } \nu_T = +1 \\ \frac{m-|A|+|A\cap T^+|}{m-1} & \text{if } \nu_T = -1 \end{cases}.$$

Recall that T^+, T^- and T^* denote the hypotheses with $+1$, -1 and $*$ outcomes for test T. So,

$$\frac{f(\nu_E \cup \{\nu_T\}) - f(\nu_E)}{1 - f(\nu_E)} = \begin{cases} \frac{|A\cap T^-|}{|A|-1} & \text{if } \nu_T = +1 \\ \frac{|A\cap T^+|}{|A|-1} & \text{if } \nu_T = -1 \end{cases}.$$

It is then straightforward to verify the following.

Proposition 2. *Consider implementing Algorithm 2 on an ODTN instance. Suppose after selecting tests E, the expanded-scenarios H' (and original scenarios A) are compatible with the parameters described above. For any test T, if $b_T \in \{+1, -1\}$ is the outcome corresponding to $B_T(H')$ then the second term in* $\mathrm{Score}_c(T; E, H')$ *and* $\mathrm{Score}_r(T; E, H')$ *is:*

$$\left(\frac{|A\cap T^-|}{|A|-1} + \frac{|A\cap T^+|}{|A|-1}\right)\cdot\frac{p(A\cap T^*)}{2} + \frac{|A\cap T^-|}{|A|-1}\cdot p\left(A\cap T^+\right) + \frac{|A\cap T^+|}{|A|-1}\cdot p\left(A\cap T^-\right).$$

The above expression has a natural interpretation for ODTN: conditioned on the outcomes ν_E so far, it is the expected number of newly eliminated hypotheses due to test T (normalized by $|A|-1$).

The first term of the score $\pi\left(L_T\left(H'\right)\right)$ or $\pi\left(R_T\left(H'\right)\right)$ is calculated as for the general ASRN problem. Finally, observe that for the submodular functions used for ODTN, the separation parameter is $\varepsilon = \frac{1}{m-1}$. So, by Theorem 7 we immediately obtain a polynomial time $O(\min(r, c) + \log m)$-approximation for ODTN.

II.7. ODTN with Many Unknowns

Our adaptive algorithm in Section II.6 has a performance guarantee that grows with the noise *sparsity* $\min(r, c\log|\Omega|)$. In this section, we consider the special case of ODTN (which is our primary application) and focus on instances with a large number of noisy outcomes. We show that an $O(\log m)$-approximation algorithm can be achieved even in this regime.

An ODTN instance is called *α-sparse* $(0 \le \alpha \le 1)$ if there $\max\{|T^+|, |T^-|\} \le m^\alpha$ for all tests $T \in \mathcal{T}$. In particular, when $\alpha < 1$, this means the vast majority of entries are noisy in every test. Our main result is the following.

Theorem 8. *There is a polynomial time adaptive algorithm whose cost is $O(\log m)$ times the optimum for ODTN on any α-sparse instance with $\alpha \le \frac{1}{2}$, and returns the true hypothesis with probability $1 - m^{-1}$.*

Moreover, by repeating the algorithm for $c \ge 1$ times, the error probability will decrease to m^{-c}.

II.7.1. Stochastic Set Cover Problem Stochastic Set Cover.

The design and analysis of our algorithm are both closely related to that of the *Stochastic Set Cover* (SSC) problem (Liu et al. (2008), Im et al. (2016)). An instance of SSC consists of a *ground set* $[m]$ of *items* and a collection of *random* subsets $S_1, \cdots, S_n$ of $[m]$, where the distribution of each S_i is known to the algorithm. The instantiation of each set is only known after it is selected. The goal is to find an adaptive policy that minimizes the expected number of sets to cover all elements in the ground set.

The following natural adaptive greedy algorithm is known to be an $O(\log m)$-approximation (Liu et al. (2008), Im et al. (2016)). Suppose at some iteration, $A \subseteq [m]$ is the set of uncovered elements. A random set S is said to be *β-greedy* if its expected coverage of the uncovered elements is at least $1/\beta$ the maximum, i.e.

$$\mathbb{E}\big[|S \cap A|\big] \ge \frac{1}{\beta} \max_{j \in [n]} \mathbb{E}\big[|S_j \cap A|\big].$$

An SSC algorithm is *(β, ρ)-greedy* if for every $t \ge 1$, the algorithm picks a β-greedy set in no less than t/ρ iterations among the first t. By slightly modifying the analysis in Im et al. (2016), one may obtain the following guarantee which will serve as the cornerstone of our analysis.

Theorem 9 **(Im et al. (2016))**. *For any stochastic set cover instance, a (β, ρ)-greedy policy costs at most $O(\beta\rho\log m)$ times the optimum.*

Relating ODTN Optimum and SSC: A Lower Bound. We now derive a lower bound on the ODTN optimum, in terms of the optima of SSC instances constructed as follows. For any hypothesis $i \in [m]$, let SSC(i) denote the stochastic set cover instance with ground set $[m] \setminus \{i\}$ and n random sets, given by

$$S_T(i) = \begin{cases} T^+ \text{ with prob. } 1 & \text{if } i \in T^- \\ T^- \text{ with prob. } 1 & \text{if } i \in T^+ \\ T^- \text{ or } T^+ \text{ with prob. } \frac{1}{2} \text{ each} & \text{if } i \in T^* \end{cases}, \qquad \forall T \in [n].$$

To see the connection between SSC and ODTN, observe that when i is the target hypothesis in the ODTN instance, any feasible algorithm must identify i by *eliminating* all other hypotheses which, in the language of SSC, translates to *covering* all items in $[m]\backslash\{i\}$. This leads to the following key lower bound that our algorithm exploits.

Lemma 8. $\mathrm{OPT} \geq \sum_{i\in[m]} \pi_i \cdot \mathrm{OPT}_{\mathrm{SSC}(i)}$.

Proof. Consider any feasible decision tree $\mathbb{T}$ for the ODTN instance and any hypothesis $i \in [m]$. If we *condition* on $\bar{i} = i$ then $\mathbb{T}$ corresponds to a feasible adaptive policy for $SSC(i)$. This is because:

- for any expanded hypothesis $(\omega, i) \in \Omega(i)$, the tests performed in $\mathbb{T}$ must rule out all the hypotheses $[m]\backslash i$, and
- the hypotheses ruled-out by any test T (conditioned on $\bar{i} = i$) is a random subset that has the same distribution as $S_T(i)$.

Formally, let $P_{i,\omega}$ denote the path traced in $\mathbb{T}$ under test outcomes ω, and $|P_{i,\omega}|$ the number of tests performed along this path. Recall that u_i is the number of unknown tests for i, and that the probability of observing outcomes ω when $\bar{i} = i$ is 2^{-u_i}, so this policy for $SSC(i)$ has cost $\sum_{(i,\omega)\in\Omega(i)} 2^{-u_i} \cdot |P_{i,\omega}|$. Thus, $OPT_{SSC(i)} \leq \sum_{(i,\omega)\in\Omega(i)} 2^{-u_i} \cdot |P_{i,\omega}|$. Taking expectations over $i \in [m]$ the lemma follows. □

We now explain why "good" progress made in SSC(i) also leads to "good" progress in ODTN. Consider a hypothesis i and a test T with $i \in T^*$, and let A be the set of consistent hypotheses. When test T is selected, the expected coverage of the corresponding (random) set $S_T(i)$ in SSC(i) is $\frac{1}{2}(|T^+ \cap A| + |T^- \cap A|)$. The following result shows that if T maximizes $\frac{1}{2}(|T^+ \cap A| + |T^- \cap A|)$, then it is 2-greedy for SSC(i).

Lemma 9. *Let T be a test that maximizes $\frac{1}{2}(|T^+ \cap A| + |T^- \cap A|)$. Then for any $i \in T^*$,*

$$\frac{1}{2}\left(|T^+ \cap A| + |T^- \cap A|\right) = \mathbb{E}\left[|S_T(i) \cap (A\backslash i)|\right] \geq \frac{1}{2} \cdot \max_{T'\in[n]} \mathbb{E}\left[|S_{T'}(i) \cap (A\backslash i)|\right].$$

Proof. For simplicity write $(T')^+$ as T'_+ (similarly define T'_-, T'_*). Note that $\mathbb{E}[|S_T(i) \cap (A\backslash i)|] = \frac{1}{2}(|T^+ \cap A| + |T^- \cap A|)$ because $i \in T^*$. We consider two cases for test $T' \in \mathcal{T}$.

- If $M_{T',i} = *$, then
$$\mathbb{E}[|S_{T'}(i) \cap (A \backslash i)|] = \frac{1}{2}\left(|T'_+ \cap A| + |T'_- \cap A|\right) \leq \frac{1}{2}\left(|T^+ \cap A| + |T^- \cap A|\right),$$
by the "greedy choice" of T in step 7.
- If $i \in T'_+ \cup T'_-$ then
$$\mathbb{E}[|S_{T'}(i) \cap (A \backslash i)|] \leq \max\{|T'_+ \cap A|, |T'_- \cap A|\} \leq |T'_+ \cap A| + |T'_- \cap A|,$$
which is at most $|T^+ \cap A| + |T^- \cap A|$ by the choice of T.

In either case the claim holds, and the lemma follows. □

Hence, by our sparsity assumption, since the vast majority of hypotheses are in T^*, such a test T is 2-greedy for most SSC instances. This motivates the following greedy algorithm. When A is the set of consistent hypotheses, pick test T that maximizes $\frac{1}{2}|T^+ \cap A| + \frac{1}{2}|T^- \cap A|$. Suppose the following *ideal condition* holds. At each iteration t (when t tests have been selected), for *every* hypothesis i, the algorithm has selected *at least* t/ρ tests that are $\star$-tests for i. Then, the sequence of tests selected is $(2, \rho)$-greedy for *every* i, hence making nearly-optimal progress in *every* instance SSC(i). Therefore by Theorem 9, the expected cost of this algorithm under i is $O(\rho \log m) \cdot \mathrm{OPT}_{\mathrm{SSC}}(i)$. Taking expectation over the target hypothesis i and combining with Lemma 8, it then follows that this algorithm is an $O(\rho \log m)$-approximation to ODTN.

However, in general, the ideal condition assumed above may not hold. In other words, up until some point, the sequence of tests selected is no longer $(2, \rho)$-greedy for some hypothesis i. To handle this issue, we modify the above greedy algorithm at all *power-of-two* iterations as follows (see Section II.7.3). At each $t = 2^k$ where $k = 1, 2, \ldots \log m$, we consider the set Z of $O(m^\alpha)$ hypotheses with the fewest $\star$-tests selected thus far. Then, we invoke a *membership oracle* Member(Z), to check whether the target hypothesis $\bar{i} \in Z$ (see Section II.7.2). If so, then the algorithm halts and returns $\bar{i}$. Otherwise, it continues with the greedy algorithm until the next power-of-two iteration. We will show that the membership oracle only incurs cost $O(m^\alpha)$, which can be bounded using the following lower bound.

Lemma 10. *The optimal value* $\mathrm{OPT} \geq \Omega(m^{1-\alpha})$ *for any α-sparse instance.*

Proof. By definition of α-sparse instances, the maximum number of candidate hypotheses that can be eliminated after performing a single test is m^α. As we need to eliminate $m - 1$ hypotheses irrespective of the realized hypothesis $\bar{i}$, we need to perform at least $\frac{m-1}{m^\alpha} = \Omega(m^{1-\alpha})$ tests under every $\bar{i}$, and the proof follows. □

In particular, when $\alpha < \frac{1}{2}$ the above implies that the cost $O(m^{\alpha})$ for each call of the membership oracle is lower than OPT, and hence the total cost incurred at power-of-two steps is $O(\log m \cdot \mathrm{OPT})$.

II.7.2. Membership Oracle The *membership oracle* $\mathrm{Member}(Z)$ takes a (small) subset $Z \subseteq [m]$ as input, and decides whether the target hypothesis $\bar{i} \in Z$. At a high level, $\mathrm{Member}(Z)$ works as follows. Whenever $|Z| \geq 2$, we pick an arbitrary pair (j,k) of hypotheses in Z and let them "duel" (i.e. choose a test T with $M_{T,j} = -M_{T,k}$) until there is only a unique *survivor* i.

Let $i \in [m]$ be an arbitrary hypothesis. We show that if $\bar{i} \neq i$ then with high probability we can rule out i using very few tests. In fact, we first select an arbitrary set W of $4\log m$ deterministic tests for i, and let Y be the set of consistent hypotheses after performing these tests. Without loss of generality, we assume $i \in T^+$ for all $T \in W$. There are three cases:

- **Trivial Case:** if $\bar{i} \in T^-$ for *some* $T \in W$, then we rule out i when any test T is performed.
- **Good Case:** if $\bar{i} \in T^*$ for more than half of the tests T in W, then by Chernoff's inequality, with high probability we observe at least one "-", hence ruling out i.
- **Bad Case:** if $\bar{i} \in T^+$ for less than half of the tests T in W, then concentration bounds can not ensure a high enough probability for ruling out i. In this case, we let each hypothesis in Y *duel* with i until either i loses a duel or wins all the duels. This takes $|Z|-1$ iterations.

We formalize the above ideas in the Algorithm 4, and prove bound the cost of $\mathrm{Member}(Z)$ as follows.

Note that Steps 3, 9 and 18 are well-defined because the ODTN instance is assumed to be identifiable. If there is no new test in Step 3 with $T^+ \cap Z' \neq \emptyset$ and $T^- \cap Z' \neq \emptyset$, then we must have $|Z'| = 1$. If there is no new test in Step 9 with $z \notin T^*$ then we must have identified z uniquely, i.e. $Y = \emptyset$. Finally, in step 18, we use the fact that there are tests that deterministically separate every pair of hypotheses.

Lemma 11. *If* $\bar{i} \in Z$*, then* $\mathrm{Member}(Z)$ *declares* $\bar{i} = i$ *with probability one; otherwise, it declares* $\bar{i} \notin Z$ *with probability at least* $1 - m^{-2}$*. Moreover, the expected cost of* $\mathrm{Member}(Z)$ *is* $O(|Z| + \log m)$*.*

Proof. If $\bar{i} \in Z$ then it is clear that $i = \bar{i}$ in step 6 and $\mathrm{Member}(Z)$ declares $\bar{i} = i$. Now consider the case $\bar{i} \notin Z$. Recall that $i \in Z$ denotes the unique hypothesis that is still compatible in step 6, and that Y denotes the set of compatible hypotheses among $[m] \setminus \{i\}$, so it always contains $\bar{i}$. Hence, $Y \neq \emptyset$ in step 14, which implies that $k = 4\log m$. Also recall the definition of set S and J from (12).

Algorithm 4 Member(Z) oracle that checks if $\bar{i} \in Z$.

```
1:  Initialize: $Z' \leftarrow Z$.
2:  while $|Z'| \geq 2$ do        % While-loop 1: Finding a suspect – reducing $|Z'|$ to 1
3:      Choose any new test $T \in \mathcal{T}$ with $T^+ \cap Z' \neq \emptyset$ and $T^- \cap Z' \neq \emptyset$, observe outcome $\omega_T \in \{\pm 1\}$.
4:      Let $R$ be the set of hypotheses ruled out, i.e. $R = \{j \in [m] : M_{T,j} = -\omega_T\}$.
5:      Let $Z' \leftarrow Z' \backslash R$.
6:  Let $z$ be the unique hypothesis when the while-loop ends.        ▷ Identified a "suspect".
7:  Initialize $k \leftarrow 0$ and $Y = H$.
8:  while $Y \neq \emptyset$ and $k \leq 4 \log m$ do        ▷ While-loop 2: choose deterministic tests for $z$.
9:      Choose any new test $T$ with $M_{T,i} \neq *$ and observe outcome $\omega_T \in \{\pm 1\}$.
10:     if $\omega_T = -M_{T,i}$ then        ▷ $i$ ruled out.
11:         Declare "$\bar{i} \notin Z$" and stop.
12:     else
13:         Let $R$ be the set of hypotheses ruled out, $Y \leftarrow Y \setminus R$ and $k \leftarrow k + 1$.
14: if $Y = \emptyset$ then
15:     Declare "$\bar{i} = i$" and terminate.
16: else
17:     Let $W \subseteq \mathcal{T}$ denote the tests performed in step 9 and        ▷ Now consider the"bad" case.
            $$\begin{aligned} J &= \{j \in Y : M_{T,j} = M_{T,i} \text{ for at least } 2\log m \text{ tests } T \in W\} \\ &= \{j \in Y : M_{T,j} = * \text{ for at most } 2\log m \text{ tests } T \in W\}. \end{aligned} \tag{12}$$
18:     For each $j \in J$, choose a test $T = T(j) \in \mathcal{T}$ with $M_{T,j}, M_{T,i} \neq *$ and $M_{T,j} = -M_{T,i}$
19:     let $W' \subseteq \mathcal{T}$ denote the set of these tests.
20: if no tests in $W \cup W'$ rule out $i$ then        ▷ Let $i$ duel with hypotheses in $J$.
21:     Declare "$\bar{i} = i$".
22: else
23:     Declare "$\bar{i} \notin Z$".
```

- Case 1. If $\bar{i} \in J$ then we will identify correctly that $\bar{i} \neq i$ in step 20 as one of the tests in W' (step 18) separates $\bar{i}$ and i deterministically. So in this case we will always declare $\bar{i} \notin Z$.
- Case 2. If $\bar{i} \notin J$, then by definition of J, we have $\bar{i} \in T^*$ for at least $2 \log m$ tests $T \in W$. As i has a deterministic outcome for each test in W, the probability that all outcomes in W are

consistent with i is at most m^{-2}. So with probability at least $1-m^{-2}$, some test in W must have an outcome (under $\bar{i}$) inconsistent with i, and based on step 20, we would declare $\bar{i} \notin Z$. In order to bound the cost, note that the number of tests performed are at most: $|Z|$ in step 3, $4\log m$ in step 9 and $|J| \leq |Z|$ in step 18, and the proof follows. □

II.7.3. The Main Algorithm The overall algorithm is given in Algorithm 5. The algorithm maintains a subset of consistent hypotheses, and iteratively computes the greediest test, as formally specified in Step 7. At each $t = 2^k$ where $k = 1, 2, ... \log m$, we invoke the membership oracle.

Algorithm 5 Main algorithm for large number of noisy outcomes

```
1: Initialization: consistent hypotheses $A \leftarrow [m]$, weights $w_i \leftarrow 0$ for $i \in [m]$, iteration index $t \leftarrow 0$
2: while $|A| > 1$ do
3:     if $t$ is a power of 2 then
4:         Let $Z \subseteq A$ be the subset of $2m^\alpha$ hypotheses with lowest $w_i$
5:         Invoke Member($Z$)
6:         If a hypothesis is identified in $Z$, then Break
7:     Select a test $T \in \mathcal{T}$ maximizing $\frac{1}{2}(|T^+ \cap A| + |T^- \cap A|)$ and observe outcome $o_T$
8:     Set $R \leftarrow \{i \in [m] : M_{T,i} = -o_T\}$ and $A \leftarrow A \backslash R$        ▷ Remove incompatible hypotheses
9:     Set $w_i \leftarrow w_i + 1$ for each for each $i \in T^*$        ▷ Update the weights of the hypotheses in $T^*$
10:    $t \leftarrow t + 1$.
```

Truncated Decision Tree. Let $\mathbb{T}$ denote the decision tree corresponding to our algorithm. We only consider tests that correspond to step 7. Recall that H is the set of *expanded* hypotheses and that any expanded hypothesis traces a unique path in $\mathbb{T}$. For any $(i,\omega) \in H$, let $P_{i,\omega}$ denote this path traced; so $|P_{i,\omega}|$ is the number of tests performed in Step 7 under (i,ω). We will work with a truncated decision tree $\overline{\mathbb{T}}$, defined below.

Fix any expanded hypothesis $(i,\omega) \in H$. For any $t \geq 1$, let $\theta_{i,\omega}(t)$ denote the fraction of the first t tests in $P_{i,\omega}$ that are $\star$-tests for hypothesis i. Recall that $P_{i,\omega}$ only contains tests from Step 7. Let $\rho = 4$ and define

$$t_{i,\omega} = \max\left\{t \in \{2^0, 2^1, \cdots, 2^{\log m}\} \;:\; \theta_{i,\omega}(t') \geq \frac{1}{\rho} \text{ for all } t' \leq t\right\}. \tag{13}$$

If $t_{i,\omega} > |P_{i,\omega}|$ then we simply set $t_{i,\omega} = |P_{i,\omega}|$.

Now we define the *truncated* decision tree $\overline{\mathbb{T}}$. By abuse of notation, we will use $\theta_i(t)$ and t_i as *random variables*, with randomness over ω. Observe that for any (i,ω), at the next power-of-two step[†] $2^{\lceil \log t_i \rceil}$, which we call the *truncation time*, the membership oracle will be invoked. Moreover, $2^{\lceil \log t_i \rceil} \le 2t_i$, . This motivates us to define $\overline{\mathbb{T}}$ is the subtree of $\mathbb{T}$ consisting of the first $2^{\lceil \log t_{i,\omega} \rceil}$ tests along path $P_{i,\omega}$, for each $(i,\omega) \in H$. Under this definition, the cost of Algorithm 5 clearly equals the sum of the cost the truncated tree and cost for invoking membership oracles.

Our proof proceeds by bounding the cost of Algorithm 5 at power-of-two steps and other steps. In other words, we will decompose the cost into the cost incurred by invoking the membership oracle and selecting the greedy tests. We start with the easier task of bounding the cost for the membership oracle. The oracle Member is always invoked on $|Z| = O(m^\alpha)$ hypotheses. Using Lemma 11, the expected total number of tests due to Step 4 is $O(m^\alpha \log m)$. By Lemma 10, when $\alpha \le \frac{1}{2}$, this cost is $O(\log m \cdot \mathrm{OPT})$.

The remaining part of this subsection focuses on bounding the cost of the truncated tree as $O(\log m) \cdot \mathrm{OPT}$. With this inequality, we obtain an expected cost of

$$O(\log m) \cdot (m^\alpha + OPT) \le_{(\text{as } \alpha < \frac{1}{2})} O(\log m) \cdot (m^{1-\alpha} + OPT) \le_{\big(\text{Lemma } 10\big)} O(\log m) \cdot OPT,$$

and Theorem 8 follows. At a high level, for a fixed hypothesis $i \in [m]$, we will bound the cost of the truncated tree as follows:

$$\begin{aligned} & i \text{ has low fraction of } \star\text{-tests at } t_i \\ \underset{Lemma\ 12}{\Longrightarrow} \; & i \text{ is among the top } O(m^\alpha) \text{ hypotheses at } t_i \\ \underset{Lemma\ 11}{\Longrightarrow} \; & i \text{ is identified w.h.p. by Member}(Z) \text{ at } 2^{\lceil \log t_i \rceil} \le 2t_i, \text{ hence the truncated path is } (2,2)\text{-greedy} \\ \underset{Theorem\ 9}{\Longrightarrow} \; & \text{ the expected cost conditional on } i \text{ is } O(\log m) \cdot \mathrm{SSC}(i) \end{aligned}$$

and finally by summing over $i \in [m]$, it follows from Lemma 8 that the cost of the *truncated* tree is $O(\log m) \cdot \mathrm{OPT}$. We formalize each step below.

Consider the first step, formally we show that if $\theta_i(t) < \frac{1}{4}$, then there are $O(m^\alpha)$ hypotheses with fewer $\star$-tests than i. Suppose i is the target hypothesis and $\theta_i(t)$ drops below $\frac{1}{4}$ at t, that is, only less than a quarter of the tests selected are 2-greedy for SSC(i). Recall that if $i \in T^*$ where T maximizes $\frac{1}{2}(|A \cap T^+| + |A \cap T^-|)$, then $S_T(i)$ is 2-greedy set for SSC(i), so we deduce that less than a $\frac{t}{4}$ tests selected are $\star$-tests for i, or, at least $\frac{3t}{4}$ tests selected thus far are *deterministic* for i. We next utilize the sparsity assumption to show that there can be at most $O(m^\alpha)$ such hypotheses.

Lemma 12. *Consider any $W \subseteq \mathcal{T}$ and $I \subseteq [m]$. For $i \in I$, let $D(i) = |\{T \in W : M_{T,i} \ne *\}|$ denote the number of tests in W for which i has deterministic (i.e. ± 1) outcomes. For each $\kappa \ge 1$, define $I' = \{i \in I : D(i) > |W|/\kappa\}$. Then, $|I'| \le \kappa m^\alpha$.*

[†] Unless stated otherwise, we denote $\log := \log_2$.

Proof. By definition of I' and α-sparsity, it holds that

$$|I'| \cdot \frac{|W|}{\kappa} < \sum_{i \in I} D(i) = \sum_{T \in W} |\{i \in I : M_{T,i} \neq *\}| \leq |W| \cdot m^{\alpha},$$

where the last step follows since $|T^*| \leq m^{\alpha}$ for each test T. The proof follows immediately by rearranging. □

We now complete the analysis using the relation to SSC. Fix any hypothesis $i \in [m]$ and consider decision tree $\overline{\mathbb{T}}_i$ obtained by *conditioning* $\overline{\mathbb{T}}$ on $\bar{i} = i$. Lemma 9 and the definition of truncation together imply that $\overline{\mathbb{T}}_i$ is $(2,4)$-greedy for SSC(i), so by Theorem 9, the expected cost of $\overline{\mathbb{T}}_i$ is $O(\log m) \cdot \mathrm{OPT}_{\mathrm{SSC}(i)}$. Now, taking expectations over $i \in [m]$, the expected cost of $\overline{\mathbb{T}}$ is $O(\log m) \sum_{i=1}^{m} \pi_i \cdot \mathrm{OPT}_{\mathrm{SSC}(i)}$. Recall from Lemma 8 that

$$\mathrm{OPT} \geq \sum_{i \in [m]} \pi_i \cdot \mathrm{OPT}_{\mathrm{SSC}(i)},$$

and therefore the cost of $\overline{\mathbb{T}}$ is $O(\log m) \cdot \mathrm{OPT}$.

Correctness. We finally show that our algorithm identifies the target hypothesis $\bar{i}$ with high probability. By definition of t_i, where the path is truncated, $\bar{i}$ has less than $\frac{1}{4}$ fraction of $\star$-tests. Thus, at iteration $2^{\lceil \log t_{\bar{i}} \rceil}$, i.e. the first time the membership oracle is invoked after t_i, $\bar{i}$ has less than $\frac{1}{2}$ fraction of $\star$-tests. Hence, by Lemma 12, $\bar{i}$ is among the $O(m^{\alpha})$ hypotheses with fewest $\star$-tests. Finally it follows from Lemma 11 that $\bar{i}$ is identified correctly with probability at least $1 - \frac{1}{m}$.

II.8. Extension to Non-identifiable ODT Instances

Previous work on ODT problem usually imposes the following *identifiability* assumption (e.g. Kosaraju et al. (1999)): for every pair hypotheses, there is a test that distinguishes them deterministically. However in many real world applications, such assumption does not hold. Thus far, we have also made this identifiability assumption for ODTN (see §II.4.1). In this section, we show how our results can be extended also to non-identifiable ODTN instances.

To this end, we introduce a slightly different stopping criterion for non-identifiable instances. (Note that is is no longer possible to stop with a unique compatible hypothesis.) Define a *similarity graph* G on m nodes, each corresponding to a hypothesis, with an edge (i,j) if there is *no* test separating i and j deterministically. Our algorithms' performance guarantees will now also depend on the maximum degree d of G; note that $d = 0$ in the perfectly identifiable case. For each hypothesis $i \in [m]$, let $D_i \subseteq [m]$ denote the set containing i and all its neighbors in G. We now define two stopping criteria as follows:

- The *neighborhood* stopping criterion involves stopping when the set K of compatible hypotheses is contained in *some* D_i, where i might or might not be the true hypothesis $\bar{x}$.
- The *clique* stopping criterion involves stopping when K is contained in some clique of G.

Note that clique stopping is clearly a stronger notion of identification than neighborhood stopping. That is, if the clique-stopping criterion is satisfied then so is the neighborhood-stopping criterion. We now obtain an adaptive algorithm with approximation ratio $O(d+\min(h,r)+\log m)$ for clique-stopping as well as neighborhood-stopping.

Consider the following two-phase algorithm. In the first phase, we will identify some subset $N \subseteq [m]$ containing the realized hypothesis $\bar{i}$ with $|N| \le d+1$. Given an ODTN instance with m hypotheses and tests $\mathcal{T}$ (as in §II.4.1), we construct the following ASRN instance with hypotheses as scenarios and tests as elements (this is similar to the construction in §II.4.3). The responses are the same as in ODTN: so the outcomes $\Omega = \{+1,-1\}$. Let $U = \mathcal{T} \times \{+1,-1\}$ be the element-outcome pairs. For each hypothesis $i \in [m]$, we define a submodular function:

$$\widetilde{f}_i(S) = \min \left\{ \frac{1}{m-d-1} \cdot \Big| \bigcup_{T:(T,+1)\in S} T^- \bigcup \bigcup_{T:(T,-1)\in S} T^+ \Big| \, , 1 \right\}, \quad \forall S \subseteq U.$$

It is easy to see that each function $\widetilde{f}_i : 2^U \to [0,1]$ is monotone and submodular, and the separability parameter $\varepsilon = \frac{1}{m-d-1}$. Moreover, $\widetilde{f}_i(S) = 1$ if and only if at least $m-d-1$ hypotheses are incompatible with at least one outcome in S. Equivalently, $\widetilde{f}_i(S) = 1$ iff there are at most $d+1$ hypotheses compatible with S. By definition of graph G and max-degree d, it follows that function $\widetilde{f}_i$ can be covered (i.e. reaches value one) irrespective of the noisy outcomes. Therefore, by Theorem 7 we obtain an $O(\min(r,c)+\log m)$-approximation algorithm for this ASRN instance. Finally, note that any feasible policy for ODTN with clique/neighborhood stopping is also feasible for this ASRN instance. So, the expected cost in the first phase is $O(\min(r,c)+\log m) \cdot OPT$.

Then, in the second phase, we run a simple splitting algorithm that iteratively selects any test T that splits the current set K of consistent hypotheses (i.e., $T^+ \cap K \neq \emptyset$ and $T^- \cap K \neq \emptyset$). The second phase continues until K is contained in (i) some clique (for clique-stopping) or (ii) some subset D_i (for neighborhood-stopping). Since the number of consistent hypotheses $|K| \le d+1$ at the start of the second phase, there are at most d tests in this phase. So, the expected cost is at most $d \le d \cdot OPT$. Combining both phases, we obtain the following.

Theorem 10. *There is an adaptive $O(d+\min(c,r)+\log m)$-approximation algorithm for ODTN with the clique-stopping or neighborhood-stopping criterion.*

II.9. Experiments

We implemented our algorithms on real-world and synthetic data sets. We compared our algorithms' cost (expected number of tests) with an information theoretic lower bound on the optimal cost and show that the difference is negligible. Thus, despite our logarithmic approximation ratios, the practical performance is much better.

Chemicals with Unknown Test Outcomes. We considered a data set called WISER[‡], which includes 414 chemicals (hypothesis) and 78 binary tests. Every chemical has either positive, negative or unknown result on each test. The original instance (called WISER-ORG) is not identifiable: so our result does not apply directly. Our result can also be extended to such "non-identifiable" ODTN instances (this requires a more relaxed stopping criterion defined on the "similarity graph"). In addition, we also generated a modified dataset by removing chemicals that are not identifiable from each other, to obtain a perfectly identifiable dataset (called WISER-ID). In generating the WISER-ID instance, we used a greedy rule that iteratively drops the highest-degree hypothesis in the similarity graph until all remaining hypotheses are uniquely identifiable. WISER-ID has 255 chemicals.

Random Binary Classifiers with Margin Error. We construct a dataset containing 100 two-dimensional points, by picking each of their attributes uniformly in $[-1000, 1000]$. We also choose 2000 random triples (a, b, c) to form linear classifiers $\frac{ax+by}{\sqrt{a^2+b^2}} + c \leq 0$, where $a, b \sim N(0, 1)$ and $c \sim U(-1000, 1000)$. The point labels are binary and we introduce noisy outcomes based on the distance of each point to a classifier. Specifically, for each threshold $d \in \{0, 5, 10, 20, 30\}$ we define dataset CL-d that has a noisy outcome for any classifier-point pair where the distance of the point to the boundary of the classifier is smaller than d. In order to ensure that the instances are perfectly identifiable, we remove "equivalent" classifiers and we are left with 234 classifiers.

Distributions. For the distribution over the hypotheses, we considered permutations of power law distribution ($\Pr[X = x; \alpha] = \beta x^{-\alpha}$) for $\alpha = 0, 0.5$ and 1. Note that, $\alpha = 0$ corresponds to uniform distribution. To be able to compare the results across different classifiers' datasets meaningfully, we considered the same permutation in each distribution.

Algorithms. We implement the following algorithms: the adaptive $O(r + \log m + \log \frac{1}{\varepsilon})$-approximation (which we denote ODTN$_r$), the adaptive $O(c \log |\Omega| + \log m + \log \frac{1}{\varepsilon})$-approximation (ODTN$_c$), the non-adaptive $O(\log m)$-approximation (Non-Adap) and a slightly adaptive version

[‡] https://wiser.nlm.nih.gov

of Non-Adap (Low-Adap). Algorithm Low-Adap considers the same sequence of tests as Non-Adap while (adaptively) skipping non-informative tests based on observed outcomes. For the non-identifiable instance (WISER-ORG) we used the $O(d+\min(c,r)+\log m+\log\frac{1}{\varepsilon})$-approximation algorithms with both *neighborhood* and *clique* stopping criteria. The implementations of the adaptive and non-adaptive algorithms are available online.[§]

Data Algorithm	WISER-ID	Cl-0	Cl-5	Cl-10	Cl-20	Cl-30
Low-BND	**7.994**	**7.870**	**7.870**	**7.870**	**7.870**	**7.870**
ODTN$_r$	8.357	7.910	7.927	7.915	7.962	8.000
ODTN$_h$	9.707	7.910	7.979	8.211	8.671	8.729
Non-Adap	11.568	9.731	9.831	9.941	9.996	10.204
Low-Adap	9.152	8.619	8.517	8.777	8.692	8.803

Table 1 Cost of Different Algorithms for $\alpha = 0$ (Uniform Distribution).

Data Algorithm	WISER-ID	Cl-0	Cl-5	Cl-10	Cl-20	Cl-30
Low-BND	7.702	7.582	7.582	7.582	7.582	7.582
ODTN$_r$	8.177	7.757	7.780	7.789	7.831	7.900
ODTN$_h$	9.306	7.757	7.829	8.076	8.497	8.452
Non-Adap	11.998	9.504	9.500	9.694	9.826	9.934
Low-Adap	8.096	7.837	7.565	7.674	8.072	8.310

Table 2 Cost of Different Algorithms for $\alpha = 0.5$.

Data Algorithm	WISER-ID	Cl-0	Cl-5	Cl-10	Cl-20	Cl-30
Low-BND	6.218	6.136	6.136	6.136	6.136	6.136
ODTN$_r$	7.367	6.998	7.121	7.150	7.299	7.357
ODTN$_h$	8.566	6.998	7.134	7.313	7.637	7.915
Non-Adap	11.976	9.598	9.672	9.824	10.159	10.277
Low-Adap	9.072	8.453	8.344	8.609	8.683	8.541

Table 3 Cost of Different Algorithms for $\alpha = 1$.

Results. Tables 1, Tables 2 and Tables 3 show the expected costs of different algorithms on all uniquely identifiable data sets when the parameter α in the distribution over hypothesis is $0, 0.5$ and 1 correspondingly. These tables also report values of an information theoretic lower bound (the entropy) on the optimal cost (Low-BND). As the approximation ratio of our algorithms

[§] https://github.com/FatemehNavidi/ODTN ; https://github.com/sjia1/ODT-with-noisy-outcomes

Data Parameters	WISER-ORG	WISER-ID	Cl-0	Cl-5	Cl-10	Cl-20	Cl-30
r	388	245	0	5	7	12	13
Avg-r	50.46	30.690	0	1.12	2.21	4.43	6.54
h	61	45	0	3	6	8	8
Avg-h	9.51	9.39	0	0.48	0.94	1.89	2.79

Table 4 Maximum and Average Number of Stars per Hypothesis and per Test in Different Datasets.

Algorithm	Neighborhood Stopping	Clique Stopping
ODTN$_r$	11.163	11.817
ODTN$_h$	11.908	12.506
Non-Adap	16.995	21.281
Low-Adap	16.983	20.559

Table 5 Algorithms on WISER-ORG dataset with Neighborhood and Clique Stopping for Uniform Distribution.

are dependent on maximum number c of unknowns per hypothesis and maximum number r of unknowns per test, we also have included these parameters as well as their average values in Table 4. Table 5 summarizes the results on WISER-ORG with clique and neighborhood stopping criteria. We can see that ODTN$_r$ consistently outperforms the other algorithms and is very close to the information-theoretic lower bound.

Chapter III Markdown Pricing Under Unknown Demand

We consider the Unimodal Multi-Armed Bandit problem where the goal is to find the optimal price under an unknown unimodal reward function, with an additional *markdown* constraint that requires that the price exploration is non-increasing. This markdown optimization problem faithfully models a single-product revenue management problem where the objective is to adaptively reduce the price over a finite sales horizon to maximize expected revenues.

We measure the performance of an adaptive exploration-exploitation policy in terms of the regret: the revenue loss relative to the maximum revenue that could have been attained when the demand (or revenue) curve is known in advance. For the case of L-Lipschitz-bounded unimodal revenue functions with infinite inventory, we presented in the last chapter a natural policy with regret $O(T^{3/4}(L\log T)^{1/4})$, as well as almost-matching lower bound of $\Omega(L^{1/4}T^{3/4})$ on the regret of any policy. Further, under mild assumptions, we show that the above tight bounds also hold when the inventory is finite but is at least $\Omega(T)$. Our tight regret bound highlight the additional complexity of the markdown constraint, and are asymptotically higher than the corresponding bounds without this markdown requirement of $\tilde{\Theta}(T^{1/2})$ for unimodal bandits and $\tilde{\Theta}(L^{1/3}T^{2/3})$ for L-Lipschitz bandits. We finally consider a generalization called Dynamic Pricing with Markup Penalty where the seller is allowed to increase the price by paying a markup penalty of magnitude $O(T^c)$ per markup where $c \in [0,1]$ is a given constant. We extend our results to a tight $\tilde{O}(T^{\mathrm{med}\{\frac{2}{3},\frac{3}{4},c\}})$ regret bound for this variant[¶].

III.1. Introduction

Consider the problem of dynamic pricing under *unknown* demand. This problem is by now well-studied, and indeed "optimal" solutions exist under numerous variations on (a) the set of demand functions allowed, on (b) how inventory is treated, and on (c) the frequency at which prices are allowed to change, just to name a few. By and large, these problems are modeled as variants of the classic *multi-armed bandit* problem, and optimality (with respect to a performance measure called *regret*) is achieved by striking a carefully-tuned balance between selecting prices to learn the unknown demand function (exploration), and prices to maximize revenue given what has previously been learned (exploitation).

Now a seemingly innocuous assumption made across all of this work, which appears to be critical in achieving meaningful results (i.e. sub-linear regret), is that the price is allowed to be both

[¶] med$\{a, b, c\}$ denotes the median of the numbers a, b, c.

decreased (*marked down*) and increased (*marked up*). In reality, markdowns are quite common, but this treatment of markups as being equally common and harmless in fact stands in contrast to the *practice* of pricing, where it is well-understood that markups negatively impact customers' perception of a product's value. As observed by Bitran and Mondschein (1997),

> *"Customers will hardly be willing to buy a product whose price oscillates, from their point of view, randomly over the season...Most retail stores do not increase the price of a seasonal or perishable product despite the fact that the product is being sold successfully."*

For this reason, *markdown pricing* (i.e. where markups are not allowed) has long been ubiquitous in retail (Petro (2017)), and remains among the standard set of capabilities that retailers are still seeking to hone – a recent survey (Google (2021)) suggests that up to \$39 billion in value is being left on the table due to sub-optimal markdown pricing, and this number is just for one of many sectors of retail ("specialty" retail).

In short, despite the rich literature on dynamic pricing under unknown demand in recent years, a basic question remains open with respect to the salient challenge of markdown pricing: **Is it feasible to achieve any meaningful performance for markdown pricing under unknown demand, and if so, what is the "separation" from ordinary dynamic pricing?** Put another way, does a markdown constraint render dynamic pricing less "effective", and if so, by how much? This work presents the first definitive answer to this basic question by providing an *optimal* policy for markdown pricing, which allows for a precise characterization of the separation between the regret bounds of markdown pricing and ordinary pricing.

III.1.1. Our Contributions. We study a canonical pricing problem with an additional *markdown constraint*. Specifically, at each of T discrete time periods, a price x is chosen and a random demand is observed whose mean is given by an unknown demand function $D(x)$. The markdown constraint precisely means that if price x is selected at time period t, then the price at time period $t+1$ can be at most x. We place only minimal assumptions on the demand function: that the corresponding revenue function $R(x) = xD(x)$ be unimodal and Lipschitz (we will see later on that both are necessary), and inventory is assumed to be infinite (though we will later relax this assumption). The goal is to design a policy which minimizes regret (defined as the difference between the policy's expected total revenue and the maximum total revenue that can be accrued).

Without the markdown constraint, this problem has previously been solved, and it has been shown that there exists a policy which achieves $O(T^{2/3})$ regret (Kleinberg (2005)). This policy selects a certain discrete subset of the prices and treats each price in this discretization as an "arm" in a classic multi-armed bandit problem. So in particular, many (approximately half) of the policy's price changes are markups, and thus the introduction of the markdown constraint seems likely to (a) necessitate a different algorithmic approach, and (b) induce a "separation" in achievable performance as alluded to above.

Against this backdrop, we make the following contributions:

1. **A Markdown Policy and Performance Guarantee:** We introduce a policy which satisfies the markdown constraint, and show (via Theorem 11) that it achieves $\tilde{O}(T^{3/4})$ regret.[‖] This immediately answers the first part of our basic question affirmatively: we *are* able to achieve meaningful performance in the form of a sub-linear (in T) regret bound. Moreover, with small but non-trivial modifications to our policy and proof technique, we show that:
   (a) We can relax the assumption of infinite inventory and still achieve the same $\tilde{O}(T^{3/4})$ regret in the regime where the inventory scales as $\Omega(T)$; see Theorem 12.
   (b) Stronger regret guarantees can be obtained if more stringent restrictions are placed on the revenue function. For example, $\tilde{O}(T^{5/7})$ regret can be achieved under twice-differentiability of the revenue function; see Theorem 14.
2. **Optimality via a Minimax Lower Bound:** We prove that our policy is in fact order-optimal by showing (via Theorem 13) that the regret of *any* policy is at least $\Omega(T^{3/4})$. This answers the second part of our question: the separation between markdown and ordinary pricing is precisely that markdown pricing must incur at least $\Omega(T^{3/4})$ regret, whereas ordinary pricing can achieve $\tilde{O}(T^{2/3})$ regret.

   Our proof uses a novel generalization of the classic Wald-Wolfowitz Theorem for hypothesis testing, which may be of independent interest for proving lower bounds for a broader class of online learning problems.
3. **Model Extension with Penalized Markups:** A natural generalization of our model would be one in which markups are allowed, but penalized. While a *complete* treatment of dynamic pricing with penalized markups would be substantial (indeed, we will see that even the choice of how to *model* these penalties is not obvious), we initiate this future direction of research by considering one version in which each markup incurs a fixed, known, additive cost that scales as $\Theta(T^c)$, for some $c \in [0,1]$. We provide a complete solution for this model, showing that:
   (a) A simple variant of the Successive Elimination Policy, a classical policy for MAB, achieves $\tilde{O}(T^{\mathrm{med}\{\frac{2}{3},c,\frac{3}{4}\}})$ regret when applied on a suitable discretization of the price space; see Theorem 17.

[‖] We use $\tilde{O}$ to hide logarithmic terms in T.

(b) This bound is optimal up to logarithmic factors; see Theorem 18.

These results completely characterize the manner in which our penalized markup model interpolates between ordinary pricing and markdown pricing. When the markup penalty is sufficiently low ($c \leq 2/3$), there is effectively no penalty for markups, since the achievable regret matches that for ordinary pricing. This is already quite surprising – for example, one corollary to this is that any sort of *one-time* or constant-sized penalty is an insignificant detractor to marking up (using carefully-constructed policies). When the penalty is sufficiently high ($c \geq 3/4$), this effectively imposes the hard markdown constraint, as it is optimal to *never* markup, and the resulting regret matches that for markdown pricing. Finally, the optimal regret interpolates smoothly between these two regimes for $c \in [\frac{2}{3}, \frac{3}{4}]$.

4. **Experimental Evaluation:** We test our policy on two of the most commonly-used families of demand functions, comparing against natural benchmarks designed specifically for these families. These experiments establish:

   (a) Fast convergence rate of regret: compared to an explore-then-commit (ETC) type policy which knows the specific functional form of the demand function, the regret of our policy vanishes at a considerable speed.

   (b) Robustness to model misspecification: our policy has vanishing regret on various families of demand functions, whereas an ETC-type policy may incur non-vanishing regret when it assumes an incorrect demand model.

The remainder of this paper is organized as follows: we conclude this section with a summary of the related literature. We then formally describe our model, assumptions, policies, and core results in Section III.2. The proofs of our upper and lower regret bounds are given in Sections III.3 and III.4, respectively. Section III.5 introduces our model and results for penalized markups. Experiments are described in Section III.6.

III.1.2. Previous Work The present work falls into two primary streams of work: dynamic pricing and multi-armed bandits. As mentioned above, the distinguishing feature of our work is the combination of a markdown constraint with a bandit-style (i.e. minimizing regret) analysis. Other important dimensions along which to contrast this work with the extant literature include: whether the underlying demand function is assumed to come from a parametric family (this work is non-parametric), whether infinite inventory is assumed (this work allows for a particular regime of finite inventory), and whether it is assumed that a prior distribution for the demand functions is given (this work does not). Table 6 summarizes the most related works along these dimensions.

	Regret	**Parametric**	**Markdown**	**∞-Inv.**	**Bayesian**
Smith and Achabal (1998)		N/A	✓		N/A
Kleinberg and Leighton (2003)	✓	No		✓	No
Besbes and Zeevi (2009)	✓	Both			No
Yin et al. (2009)		N/A	✓		N/A
Broder and Rusmevichientong (2012)	✓	Yes		✓	No
Harrison et al. (2012)		No		✓	Yes
Combes and Proutiere (2014)	✓	No		✓	No
Wang et al. (2014)	✓	Both			No
Cheung et al. (2017)	✓	No		✓	No
Ferreira et al. (2018)	✓	Yes			Yes
This work	✓	No	✓		No

Table 6 Comparison of our work with prior related work along important model dimensions: whether or not (1) the metric used is regret; (2) the given family of demand/revenue curve is parametric; (3) the markdown constraint is considered, (4) infinite inventory is assumed and (5) a prior over the demand family, over which the Bayesian regret is considered, is given.

Dynamic Pricing: Gallego and Van Ryzin (1994) characterized the optimal pricing policy when the demand function is known. Kleinberg and Leighton (2003) studied a revenue maximization problem for a seller with an unlimited supply of identical goods, and obtained tight regret bounds under different valuation models of buyers, including identical, random, worst-case. Besbes and Zeevi (2009) studied the dynamic pricing problem under finite inventory in a finite selling period. Their benchmark regret function is the optimal pricing algorithm which is non-adaptive and whose expected sales is at most the inventory level. They presented an algorithm which achieves nearly optimal regret bounds. Subsequently, Wang et al. (2014) improved their results by showing matching lower bound. Later, Babaioff et al. (2015) and Badanidiyuru et al. (2013) considered a more practical scenario where the inventory is finite. Other works that formulate dynamic pricing as MAB include Bastani et al. (2019), Hu et al. (2016), Chen and Farias (2018), Lei et al. (2014), Keskin and Zeevi (2014), den Boer and Zwart (2013), Liu and Cooper (2015), Farias and Van Roy (2010), Lobel (2020), Qiang and Bayati (2016), Papanastasiou and Savva (2017), den Boer and Zwart (2015).

In practice, costs of implementing frequent price-changes in a traditional retail setting can amount to a considerable portion of the seller's net margins. Thus motivated, Celik et al. (2009) considered the pricing problem with costly price adjustments. Later, for the setting where the demand is unknown, Broder (2011) formulated the demand learning problem with limited price changes and presented an $\tilde{O}(\sqrt{T})$ regret policy for parametric models using $O(\log T)$ price changes. Later, Perakis and Singhvi (2019) showed under stronger assumptions that the same regret may be achieved using $O(\log\log T)$ price-changes. Cheung et al. (2017) considered given discrete family of demand functions and presented a regret bound that decreases in the number of allowed price-changes. Chen et al. (2020) considered the joint pricing and inventory management problem under limited price changes.

Orthogonal to the number of price changes, previous literature has also considered the *direction* of price changes. In practice, buyers usually have a *reference price* in mind, at which a higher (lower) price is considered a loss (gain), and customers are more sensitive to losses than to gains. Dynamic pricing with reference-price effects has been studied extensively in recent years, for example Nasiry and Popescu (2011), Heidhues and Kőszegi (2014), Wu et al. (2015), Hu et al. (2016), Wang (2016), Recently, den Boer and Keskin (2020) considered the setting where the demand function is unknown.

As an important variant of the dynamic pricing problem, the *Markdown Pricing* problem has been extensively studied. The book chapter by Ramakrishnan (2012) and surveys by Elmaghraby and Keskinocak (2003) and den Boer and Zwart (2015) provide a thorough overview. Most previous work on markdown pricing assume a known demand function and focused on either empirical results (e.g. Smith and Achabal (1998), Heching et al. (2002)) or strategic customer behavior (e.g. Yin et al. (2009), Boyacı and Özer (2010), Aviv and Vulcano (2012)). However, little is known about the setting where the demand function is unknown. Birge et al. (2019) considered the markdown pricing with unknown demand and proposed a model that aims at the strategic consumer behavior in markdown pricing, and showed forward-looking customers can improve the performance of a learning policy. In contrast, in this work, we tackle the problem from a different perspective, by simply viewing it as a dynamic pricing problem with an additional monotonicity constraint.

Multi-armed Bandits (MAB): There exist several MAB variants that are similar to our problem, but without the markdown constraint. In the *Discrete Multi-armed Bandit* problem, the player is offered a finite set of arms, with each arm providing a random revenue from an unknown probability distribution specific to that arm. The objective of the player is to maximize the total revenue earned by pulling a sequence of arms (e.g. Lai and Robbins (1985)). Our pricing problem generalizes this framework by using an infinite action space $[0,1]$ with each price p corresponding to an action whose revenue is drawn from an unknown distribution with mean $R(p)$.

In the *Lipschitz Bandit* problem (e.g. Agrawal (1995)), it is assumed that each $x \in [0,1]$ corresponds to an arm with mean reward $\mu(x)$, and μ satisfies the Lipschitz condition, i.e. $|\mu(x) - \mu(y)| \leq L|x-y|$ for some constant $L > 0$. Kleinberg (2005) proved a tight $\tilde{\Theta}(L^{1/3}T^{2/3})$ regret bound for one-dimensional Lipschitz Bandits. The lower bound was proved by considering a family of "bump curves": each curve is $\frac{1}{2}$ at all arms except in a small neighborhood of the "peak", where the mean reward is elevated by a constant. Since these bump curves are unimodal, this lower bound carries over to the family we study.

Another closely-related variant of MAB is the *Unimodal Bandits* problem (Cope (2009), Yu and Mannor (2011), Combes and Proutiere (2014)). In addition to the Lipschitzness assumption, the reward function $\mu : [0,1] \to [0,1]$ is assumed to be unimodal. It is also assumed that there is a constant $L' > 0$ s.t. $|\mu(x) - \mu(y)| \geq L'|x-y|$ for all $x, y \in [0,1]$. Yu and Mannor (2011) proposed a binary-search type algorithm with regret $\tilde{O}(\sqrt{T})$.

Somewhat surprisingly, a seemingly irrelevant line of work – decision making under individual fairness constraint – turns out to be closely related to the markdown pricing problem. Due to the fairness constraint, the learner has to be cautious in the exploration phase to avoid violating the fairness constraint in the future. For instance, "it is typical for legal stances on new issues to be more conservative initially and then potentially become more liberal over time as the impact and nuances of these issues become clear" (Gupta and Kamble (2019)). The Cautious Fair Exploration policy is, in spirit, similar to our Uniform Elimination policy (Algorithm 11) for the infinite inventory scenario. Motivated by the fairness constraint, follow-up work (Salem et al. (2021)) considered a more general online convex optimization problem where the actions sequence is required to be monotone. However, their work focuses on gradient descent based algorithms for smooth concave reward functions, while our work makes much weaker assumptions on the reward functions.

Finally we note that very recently, independent of our work, Chen (2021) considered a special case where the inventory is infinite under the name *Monotone Bandits*, and obtained the same results using a different lower bound technique. Their lower bound is established through careful manipulation of basic information theoretical tools, while ours relies on a novel and powerful tool – the Wald-Wolfowitz Theorem, which we also show to be useful beyond the markdown pricing problem. Further, our work extends the idea for infinite inventory to a variety of more practical settings such as finite inventory, smooth reward functions, and dynamic pricing with markup penalties.

III.2. Model

We begin by formally stating our model. Given inventory $I > 0$ and a discrete time horizon of T rounds, in each round t, the policy (representing the "seller"), selects a price $x_t \in [0,1]$ (the particular interval $[0,1]$ is without loss of generality, by scaling). This round's demand d_t is then independently drawn from a fixed distribution with unknown mean $D(x_t)$, and the policy receives reward x_t for each unit sold (up to the smaller of the demand and remaining inventory):

$$\min\{d_t, I-(d_1+\cdots d_{t-1})\}x_t.$$

For simplicity, we will assume that the random demand d_t is almost surely bounded, specifically in $[0,1]$ (again, w.l.o.g.), though our results can be easily extended to sub-Gaussian distributions. The only constraint the policy must satisfy is the *markdown constraint*: $x_1 \geq \cdots \geq x_T$ almost surely.

The function $D(x)$ which maps each price x to the mean demand at that price is known as the *demand function*. A demand function $D(x)$ is naturally associated with a *revenue function* $R(x) = xD(x)$. For most of this paper, we will deal directly with revenue functions, which we term more generally as *reward* functions.[**] For any policy $\mathbb{A}$,[††] reward function $R(\cdot)$, and inventory I, we use $r(\mathbb{A}, R, I)$ to denote the expected total reward of $\mathbb{A}$ under R with initial inventory I.

Rather than evaluating policies directly in terms of $r(\mathbb{A}, R, I)$, it is more informative (and ubiquitous in the literature on multi-armed bandits) to measure performance using the notion of *regret* with respect to a certain idealized benchmark. Here, we will define regret with respect to the best possible *fixed* price policy. Specifically, a *Fixed Price Policy* (FPP) selects the same price at each round, i.e. $x_1 = \cdots = x_T = p$ for some p. Let $FPP(p)$ denote the FPP at price p. We use OPT_R to denote the maximum achievable expected reward among all FPPs, i.e.

$$\mathrm{OPT}_R := \max_{p\in[0,1]} r(FPP(p), R, I).$$

So for example, when $I = \infty$, we have that $\mathrm{OPT}_R = r^*T$, where $r^* = \max_{x\in[0,1]} R(x)$. The regret of a policy is then defined with respect to this quantity, and we seek to bound the *worst-case* value over a given family of reward functions.

Definition 2 (Regret). Let $\mathcal{F}$ be a family of reward functions, each a mapping from $[0,1]$ to $[0,1]$. For any policy $\mathbb{A}$ and $R \in \mathcal{F}$, define the *regret* of policy $\mathbb{A}$ under R to be

$$\mathrm{Reg}(\mathbb{A}, R, I) := \mathrm{OPT}_R - r(\mathbb{A}, R, I).$$

The *worst-case regret* of policy $\mathbb{A}$ for family $\mathcal{F}$ is $\mathrm{Reg}(\mathbb{A}, \mathcal{F}, I) := \sup_{R\in\mathcal{F}} \mathrm{Reg}(\mathbb{A}, R, I)$.

[**] The corresponding demand function can naturally be backed out from a reward function: $D(x) = R(x)/x$ for $x > 0$.

[††] For the sake of completeness, a *policy* is, formally, a time-indexed sequence of functions $\mathbb{A} = \{\mathbb{A}_t : ([0,1] \times [0,1])^{t-1} \to [0,1], t = 1, \ldots, T\}$, where each function $\mathbb{A}_t$ maps the prices selected and demands observed over the previous $t-1$ rounds to a price for round t.

To summarize, the problem we seek to solve is: given a family $\mathcal{F}$ of reward functions from $[0,1]$ to $[0,1]$ and initial inventory I, design a policy $\mathbb{A}$ that satisfies the markdown constraint and that minimizes $\text{Reg}(\mathbb{A}, \mathcal{F}, I)$. We will be particularly concerned with how a policy's regret scales with the time horizon T – at the very least, we aim for sub-linear (i.e. $o(T)$) regret.

It is worth pausing here to note that our definition of regret has different implications when I is either infinite or finite. When I is infinite, the best *offline* policy (meaning one that knows the reward function R) is precisely a fixed price policy, so regret here is really measured against the best offline policy (this is the "typical" definition of regret). However, when I is finite, the best offline policy need not be a fixed price policy, and moreover even calculating the best offline policy for a given reward function can be non-trivial – in general, the policy can at best be characterized as the solution to a dynamic program (see Talluri and Van Ryzin (2004)). Thus, we measure regret only against fixed price policies. One reason this is fairly innocuous is that for the inventory regime we consider, $I = \Omega(T)$, the best fixed price policy is asymptotically optimal (as T grows) in a manner that can be made formal.

III.2.1. Assumptions on the Reward Function We have so far made just one assumption: that the random demands are bounded (and even this can be relaxed to sub-Gaussianity). We will, in addition, require two assumptions on the underlying reward function:

1. *Lipschitz*: The reward function is L-Lipschitz, i.e. $|R(x) - R(x')| \le L|x - x'|$ for all $x, x' \in [0,1]$. This assumption is standard for the version of our problem without markdown constraint (e.g. Kleinberg (2005)). Note, as an aside, that we are implicitly assuming here that L, or at least an upper bound on L across the entire family of reward functions, is known.
2. *Unimodal:* The reward function is unimodal, i.e. there exists $x^* \in [0,1]$ s.t. R is non-increasing on $[x^*, 1]$ and non-decreasing on $[0, x^*]$. This assumption has also previously appeared for the non-markdown version of our problem (e.g. Yu and Mannor (2011)).

In addition to having appeared previously in the literature, both of these assumptions are in fact *necessary* for achieving sub-linear regret. Specifically, the Lipschitz assumption is necessary in the sense that there exists a family $\mathcal{F}$ of unimodal reward functions, whose Lipschitz constants are arbitrarily large, such that for any policy $\mathbb{A}$, its regret is $\Omega(T)$ under some $R \in \mathcal{F}$.[‡‡] The unimodal assumption is similarly necessary: there exists a family of L-Lipschitz reward functions such that any policy has regret $\Omega(T)$ under some function in the family.[§§]

[‡‡] One example family is $\mathcal{F} = \{R_c(x) = (-(x-1)(x-1+2c)/c^2)^+ : c \in (0, 1/2)\}$.

[§§] Such a family can be constructed with just *two* reward functions: one with a single mode, and one with two modes.

Finally, these two assumptions hold for the reward functions corresponding to some of the most commonly-used parametric families of demand functions:

1. *Linear Demand*: $\{D_{a,b}(x) = a - bx : 1 \geq a \geq b \geq 0\}$
2. *Exponential Demand*: $\{D_{a,b}(x) = e^{a-bx} : a \in \mathbb{R}, b \in \mathbb{R}_+\}$

These examples serve to illustrate that our assumptions are mild enough to allow for realistic models of demand, though we emphasize that our policies and results will *not* require that the reward functions be parameterizable.

III.2.2. Our Policies We can now state our policies, beginning with the setting of infinite inventory (which captures the crux of the challenge), and then finite inventory.

Infinite Inventory. Our policy under infinite inventory operates under a simple idea: begin at the highest price, and decrease the price at a constant rate, stopping when the mode (or "peak") of the reward function is detected, i.e. when the mean reward at the current price is significantly lower than some previous price. Intuitively, this idea should perform well as long as the rate of price decrease is neither too slow (or else the policy will spend too much time at sub-optimal prices before reaching the peak) nor too fast (or else the policy will not gather sufficient information to correctly identify when the peak is reached).

Algorithm 6 Uniform Elimination Policy ($\text{UE}_{s,\delta}$).

```
Input: $s, \delta, T > 0$.                                   ▷ Step size and width of target confidence intervals.
Initialize: $x \leftarrow 1$, $\text{LCB}_{\max} \leftarrow 0$, $k \leftarrow \lceil 3\delta^{-2} \log T \rceil$.
while $x > 0$ do                                             ▷ Exploration phase starts
    Select price $x$ for the next $k$ rounds and observe demands $X_1, ...X_k$.
    $\bar{\mu} \leftarrow \frac{x}{k} \sum_{i=1}^{k} X_i$                        ▷ Compute mean rewards
    $[\text{LCB}, \text{UCB}] \leftarrow [\bar{\mu} - \delta, \bar{\mu} + \delta]$.   ▷ Compute confidence interval for reward at current price
    if $\text{LCB} > \text{LCB}_{\max}$ then                              ▷ Update best LCB so far
        $\text{LCB}_{\max} \leftarrow \text{LCB}$.
    if $\text{UCB} < \text{LCB}_{\max}$ then                              ▷ Exploration phase ends
        $x_h \leftarrow x$. Break.                                ▷ Define *halting* price
    else $x \leftarrow x - s$.                                    ▷ Reduce the price by $s$
Select price $x_h$ in all future rounds.                     ▷ Exploitation phase
```

Our actual policy, dubbed the *Uniform Elimination Policy* ($\text{UE}_{s,\delta}$), implements a discretized version of this idea. As described in Algorithm 11, the policy is parameterized by two values: a *step* *size* s and a confidence interval *width* δ. Each price decrease is exactly of size s, and rather than decreasing each round, our policy remains at a price x long enough that $R(x)$ can be estimated up to an additive error of at most δ with high confidence (via Hoeffding/Chernoff bound). The policy "halts" when the confidence interval at the current price lies completely below that of a previous price, indicating that we have likely "overshot" the optimal price.

As we will see in the next subsection, this policy is order-optimal (up to logarithmic factors) for certain values of s and δ. It is important to note that these "correct" choices of s and δ depend on L and T, meaning our policy itself requires knowledge of L and T. The knowledge of L is standard (see e.g. Kleinberg (2005)), and further, in practice, one may simply choose the maximum L of fitted demand functions from past sales data as an upper bound for the Lipschitz constant of the unknown demand model. Knowledge of T is more delicate. In the literature on MAB, one of the primary challenges (and successes) has been in designing so-called *anytime* policies, which achieve order-optimal regret *without* knowledge of T. One could ask if this is possible here – this is in fact *impossible* when the markdown constraint is present (see Proposition 4).

Finite Inventory. Now assume that inventory is finite, and let $\rho := I/T$ be the inventory-to-time ratio. A simple observation allows us to modify the previous Uniform Elimination Policy for finite inventory. Fix a reward function R (and corresponding demand function D), and let p^* be the location of its peak: $p^* \in \arg\max_x R(x)$. Let p_d be the *depletion price*, meaning the value satisfying $D(p_d) = \rho$ (we assume its existence just for convenience of discussion). This is the price at which, if the demands were not random, our inventory I would be perfectly depleted after exactly T rounds.

The simple observation is that $\max\{p^*, p_d\}$ is approximately the best fixed price (we show this formally in Section III.3.2). Intuitively, if $p_d < p^*$, then there is effectively enough inventory to ignore the inventory constraint. If $p_d > p^*$, offering a price lower than p_d is sub-optimal because the same number of units would get sold (i.e. all of them) at a lower price, and offering a price higher than p_d is sub-optimal because $R(x)$ is decreasing for $x \geq p^*$.

Our *Depletion-Aware Uniform Elimination Policy* ($\text{DUE}_{s,\delta}$), described in Algorithm 7, adapts the Uniform Elimination Policy based on this observation. In particular, the Uniform Elimination Policy already seeks to decrease the price as quickly as possible until p^* is reached. An extra condition which tracks the rate at which inventory is depleted ensures that price decreases are halted if x_d is reached.

III.2.3. Our Results The core results of this paper are a set of matching upper and lower regret bounds for markdown pricing, whose ideas also lay the cornerstone for our extensions. Throughout, we use $\hat{\mathcal{F}}_L$ to denote the family of L-Lipschitz, unimodal functions from $[0,1]$ to $[0,1]$.

Algorithm 7 Depletion-Aware Uniform Elimination Policy (DUE$_{s,\delta}$).

```
Input: $s, \delta, T > 0$.                                    ▷ Step size and width of target confidence intervals
Initialize: $x \leftarrow 1, \mathrm{LCB}_{\max} \leftarrow 0$
while $x > 0$ do                                                                          ▷ Exploration Phase
    Select price $x$ in the next $k = \lceil 3\delta^{-2} \log T \rceil$ rounds (stop when inventory is depleted), and
  observe demands $X_1, ..., X_k$.
    Set $\bar{d} \leftarrow \frac{1}{k} \sum_{i=1}^{k} X_i$                                  ▷ Estimate $D(x)$
    Set $[\mathrm{LCB}, \mathrm{UCB}] \leftarrow [x\bar{d} - \delta, x\bar{d} + \delta]$      ▷ Compute confidence interval
    if $\mathrm{LCB} \geq \mathrm{LCB}_{\max}$ then                                ▷ Keep track of the highest LCB
        $\mathrm{LCB}_{\max} \leftarrow \mathrm{LCB}$.
    if $\mathrm{UCB} < \mathrm{LCB}_{\max}$ or $(\bar{d} + \delta)T \geq I$ then               ▷ Termination condition
        $x_h \leftarrow x$. Break.                                                       ▷ Exploration halts
    else
        $x \leftarrow x - s$.                                                     ▷ Reduce the price by $s$
Use price $x_h$ for all future rounds                                               ▷ Exploitation phase
```

Our first result is an upper bound on the regret of our Uniform Elimination policy for the infinite inventory setting:

Theorem 11 **(Upper Bound, Infinite Inventory)**. *For any given $L > 0$ and $T \in \mathbb{N}$, the Uniform Elimination policy* UE$_{s,\delta}$ *satisfies*

$$\mathrm{Reg}(\mathrm{UE}_{s,\delta}, \hat{\mathcal{F}}_L, I = \infty) = O(T^{3/4}(L \log T)^{1/4}),$$

for $\delta = T^{-1/4}(L \log T)^{1/4}$ and $s = \delta / L$.

The most immediate conclusion from Theorem 11 is that it establishes concretely that sub-linear regret is achievable for markdown pricing.

In practice, the initial inventory I is usually finite. Recall that $\rho := I/T$ is the inventory-time ratio. Since the range of demand functions are normalized to $[0,1]$, the seller can sell at most T units in T rounds, so the problem reduces to the $I = \infty$ case if $\rho \geq 1$. On the other extreme, if $I = o(T)$, suppose the mean demand at $p = p_{max} = 1$ is non-zero, then for any $p \in [0,1]$ the FPP is likely to sell out all units, so the optimal seller should select $p = 1$ in all rounds. Thus, the interesting scenario is $\rho = \Omega(1)$.

Theorem 12 **(Upper Bound, Finite Inventory)**. *Given any $L,T,I>0$ where $\rho=I/T=\Omega(1)$ (but not necessarily greater than 1), the Depletion-Aware Uniform-Elimination Policy* $\mathrm{DUE}_{s,\delta}$ *with for $\delta=T^{-1/4}(L\log T)^{1/4}$ and $s=\delta/L$ satisfies*

$$\mathrm{Reg}(\mathrm{DUE}_{s,\delta},\hat{\mathcal{F}}_L,I)=O\big(T^{3/4}(L\log T)^{1/4}\big).$$

To show a lower bound, we need to specify a family of problem instances on which any markdown policy suffers this amount of regret. An $\Omega(L^{1/3}T^{2/3})$ lower bound for markdown pricing with $I=\infty$ is implied by the lower bound for Lipschitz bandits (Kleinberg (2005)), since the "bump curves" they used are unimodal and Lipschitz. Despite its low-adaptivity, the DUE policy surprisingly achieves the **best possible** regret among all deterministic policies, including adaptive ones:

Theorem 13 **(Lower Bound)**. *Suppose the demand distribution at every price is Bernoulli. Then for any $L>0$, there is a family $\mathcal{M}\subset\hat{\mathcal{F}}_L$ of reward functions such that any markdown policy $\mathbb{A}$ (that knows L,T) suffers regret*

$$\mathrm{Reg}(\mathbb{A},\mathcal{M},I=\infty)=\Omega(L^{1/4}T^{3/4}).$$

Generalized Wald-Wolfowitz Theorem (GWW). The existing lower bound techniques for MAB fail to address the extra complexity caused by the markdown constraint. We develop a novel technique to address this challenge and obtain Theorem 13. We generalize the classic *Wald-Wolfowitz Theorem* (WW) for sequential hypothesis testing from testing between point estimates to testing between appropriately defined intervals. The basic idea is to construct a family of reward functions each having a unique optimal price, then use GWW to prove that any low-regret policy has to spend **in expectation** at least certain number of rounds to distinguish between each pair of reward functions, whose maxima occur nearby. As a result, if the optimal price is small, a high regret is incurred since the policy must "waste" too much time in suboptimal prices.

	Upper Bound	**Lower Bound**
Unimodal Bandits	$O(\sqrt{T})$	Unknown
Lipschitz Bandits	$O(T^{2/3}(L\log T)^{1/3})$	$\Omega(T^{2/3}L^{1/3})$
Markdown Pricing	$O(T^{3/4}(L\log T)^{1/4})$	$\Omega(T^{3/4}L^{1/4})$

Table 7 Distribution independent regret bounds for markdown pricing and related bandit problems. Our results (in red) are presented in Theorem 11 and 13. The lower bound for Lipschitz bandits is from Kleinberg (2005). Note that all but the last row hold for both known or unknown T.

Table 7 summarizes the relevant previous results, along with our new results.[¶¶] The key observation is, with the markdown constraint, the regret bounds increase significantly, which matches our intuition. For unimodal bandits, one can apply a natural binary search type algorithm (see Yu and Mannor (2011)) to localize a mode with arbitrary precision. However, such a policy cannot be extended to our setting due to the markdown constraint.

III.3. Proof of Upper Bounds

III.3.1. Infinite Inventory: Proof of Theorem 11 Theorem 11 is immediately implied by the following lemma when $\delta = \sqrt{2}T^{-1/4}(L\log T)^{1/4}$ and $s = \delta/2L$, which minimize the term inside big-O in (14). (In fact, let $g(s,\delta) = (\delta + sL)T + s^{-1}\delta^{-2}\log T$. Then, $\frac{\partial g}{\partial \delta} = T - 2\delta^{-3}s^{-1}\log T$ and $\frac{\partial g}{\partial s} = LT - s^{-2}\delta^{-2}\log T$. Hence, $\nabla g = 0 \iff s\delta^3 = \frac{2\log T}{T}$ and $s^2\delta^2 = \frac{\log T}{LT}$. One can then easily verify that our choices of s,δ satisfies the above condition and is indeed a global minimum of g.)

Lemma 13. *For any $s,\delta > 0$, it holds that*

$$\mathrm{Reg}(\mathrm{UE}_{s,\delta}, \hat{\mathcal{F}}_L, I = \infty) = O\big((\delta + sL)T + s^{-1}\delta^{-2}\log T\big). \tag{14}$$

We state a standard result, the proof of which can be found in e.g. Vershynin (2018).

Lemma 14 **(Concentration Bounds)**. *Let $Z_1,...,Z_m$ be independent random variables supported on $[0,1]$, and $\bar{Z} = \frac{1}{m}\sum_{i=1}^m Z_i$, then for any $\delta > 0$, it holds that*

$$\mathbb{P}(|\bar{Z} - \mathbb{E}(\bar{Z})| > \delta) \le \exp(-2m\delta^2) \qquad \textit{(Hoeffding's inequality) and}$$

$$\mathbb{P}\big(|\bar{Z} - \mathbb{E}(\bar{Z})| > \delta\cdot\mathbb{E}(\bar{Z})]\big) \le \exp\left(-\frac{\mathbb{E}\left(\bar{Z}\right)}{2}\delta^2\right) \qquad \textit{(Chernoff's inequality).}$$

We use the following lemma that follows from standard tail bounds.

Lemma 15. *Define $\mathcal{C}$ to be the event that $R(x) \in [\bar{\mu} - \delta, \bar{\mu} + \delta]$ for any of the prices x selected by the policy $\mathrm{UE}_{s,\delta}$ with $s \ge 1/T$. Then the probability of its complement $\mathbb{P}[\overline{\mathcal{C}}] = O(T^{-1})$.*

Proof Fix any sample price x_i. Recall that $m := \lceil 3\delta^{-2}\log T\rceil$ and let $Z_1,...,Z_m$ be the rewards for each customer when the price is x_i, then $\mathbb{E}(\bar{Z}) = R(x_i)$. Recall that the empirical mean demand is $\bar{d}_i := \frac{1}{m}\sum_{i=1}^m Z_i$, so by Lemma 14,

$$\mathbb{P}(|R(x_i) - \bar{d}_i| \ge \delta) \le \exp(-2\cdot 3\delta^{-2}\log T\cdot\delta^2) = T^{-6}.$$

The proof completes by applying the union bound over all $O(s^{-1}) = O(T)$ prices. □

[¶¶] All asymptotic bounds with L are w.r.t. both L and T. For example, our upper bound for markdown pricing policy $\mathbb{A}$ shows that there are constants $C, T_0, L_0 > 0$ s.t. for any $L \ge L_0, T \ge T_0$, we have $\mathrm{Reg}(\mathbb{A}, \hat{\mathcal{F}}) \le CT^{3/4}(L\log T)^{1/4}$.

Furthermore, when the policy halts at x_h, the confidence interval for the reward at the price just before the halting price overlapped with that of the best confidence interval found so far centered around the reward at p^*. Hence, the reward at the halting price is at most twice the length of a confidence interval, i.e. 4δ. This gives the following lemma.

Lemma 16. *If x_h denotes the halting price of $\mathrm{UE}_{s,\delta}$ on an L-Lipschitz unimodal reward function $R \in \hat{\mathcal{F}}_L$, then conditional on event $\mathcal{C}$, $R(x_h) \geq R(p^*) - 4\delta - 2sL$.*

Proof. We first show that $x_h \leq p^*$ i.e. the policy does not stop before reaching p^*. Let $\ell = \max\{i : x_i \geq p^*\}$ be the index of the lowest sample price above p^*. Recall that h is the price-index where the halting condition in algorithm 11 is satisfied. We observe that conditional on $\mathcal{C}$, the exploration phase does not terminate before reaching p^*, i.e. $h \geq l+1$. In fact, for any i,j with $x_j > x_i \geq p^*$, by Lemma 30 it holds that $UCB(x_i) \geq R(x_i) \geq R(x_j) \geq LCB(x_j)$, so the halting condition is not satisfied at price i. Therefore, $h \geq \ell+1$. Since the policy only samples prices that are s apart, and halts only after it has overshot the interval containing p^*, we may suffer a loss in reward at the halting price of about sL where L is the Lipschitz constant translating price decrements to reward decrements.

With this observation, we only need to consider two cases of x_h:

- Suppose $h = \ell+1$. Since $p^* \in [x_{\ell+1}, x_\ell]$, by Lipschitzness of R we have $R(x_{\ell+1}) \geq r^* - Ls$. Thus the regret in the exploitation phase is at most LsT.
- Otherwise suppose $h \geq \ell+2$. Then $R(x_h)$ is almost optimal by the following lemma:

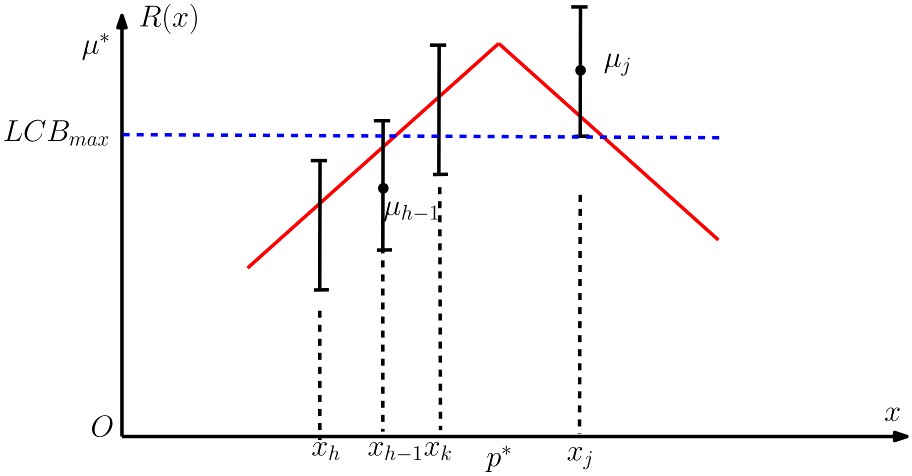

Figure 1 Illustration of the proof of Lemma 16.

Let $\bar{\mu}_i$ be the empirical mean reward at x_i (as defined in Step 6 of Algorithm 11), and $[LCB_i, UCB_i] = [\bar{\mu}_i - \delta, \bar{\mu}_i + \delta]$. We first claim that by assuming $R(x_i) \in [LCB_i, UCB_i]$ for all i, we only lose an additive $O(1)$ on regret. In fact, define B_i as the ("bad") event that $R(x_i) \notin [LCB_i, UCB_i]$, and $B = \cup_{i=1}^{\lceil 1/s \rceil} B_i$. By Lemma 30, $\mathbb{P}(B_i) = O(T^{-2})$ for each i. By the union bound, since $s \geq 2/T$, $\mathbb{P}(B) \leq T^{-2} \cdot \lceil 1/s \rceil = O(T^{-1})$. Conditional on B, the regret is $O(T)$, thus the regret contributed by B is $O(T \cdot T^{-1}) = O(1)$.

Suppose $LCB_{\max}$ be attained at x_j (See Fig 7). Then the halting condition at x_h translates to $UCB_h < LCB_j \leq UCB_{h-1}$. It follows that $|\bar{\mu}_{h-1} - \bar{\mu}_j| \leq 2\delta$, hence

$$R(x_{h-1}) \geq \bar{\mu}_{h-1} - \delta \geq (\bar{\mu}_j - 2\delta) - \delta = LCB_{\max} - 2\delta. \tag{15}$$

Next we lower bound $LCB_{\max}$. Let x_k be the first explored price lower than p^*, then

$$LCB_{\max} \geq LCB_k \geq R(x_k) - 2\delta \geq (r^* - Ls) - 2\delta. \tag{16}$$

where the last inequality follows since $|x_k - p^*| \leq s$. Finally combining the fact that $R(x_h) \geq R(x_{h-1}) - sL$ with the above,

$$\begin{aligned} R(x_h) &\geq R(x_{h-1}) - sL \\ &\geq LCB_{\max} - 2\delta - sL \\ &\geq (R(p^*) - sL - 2\delta) - 2\delta - sL \\ &= R(p^*) - 4\delta - 2sL. \qquad \square \end{aligned}$$

Proof of Lemma 13. Let R be the true reward function with an optimal price p^*. Define *sample prices* $x_i = 1 - si$ for $i \leq s^{-1}$. It follows that the regret in the exploitation phase is $O\big((\delta + sL)T\big)$. On the other hand, in each case, there are $O(\delta^{-2} \log T)$ explorations per price for up to s^{-1} different prices, giving a total of $O(s^{-1}\delta^{-2} \log T)$ rounds for exploration. Lemma 13 is proved by combining the regret in the exploitation and exploration phases. □

Discrete Prices. It is straightforward to extend this analysis to a discrete price setting and derive an upper bound in terms of the maximum gap $\Delta_{\max}$ between neighboring prices. First, we note that it is without loss of generality to assume that the *minimal* gap Δ_{min} between any two prices is at least T^{-2}. In fact, whenever there is a pair of prices at distance less than T^{-2}, we remove one of them arbitrarily. Repeating, we will obtain a subset of prices with $\Delta_{min} \geq T^{-2}$, and moreover, by doing this, we only lose an $O(T^{-1} \cdot T) = O(1)$ term in regret. Now, there are $O(T^2)$ discrete prices, so we may apply Hoeffding's bound and union bound to show that with high probability, the confidence interval at each discrete price x contains $R(x)$. We will condition on this event hereafter.

We slightly modify the UE policy as follows: in Step 12 the discrete UE policy selects the next price to be the maximum discrete price *below* $x - s$. Thereby, the regret accumulated before reaching p^* is $\sim s^{-1}\delta^{-2}\log T$. On the other side, one may easily show that the exploitation price x_{halt} satisfies $R(x_{\text{halt}}) \geq R(p^*) - O\left(\delta + (s + \Delta_{\max})L\right)$, and hence the regret after reaching p^* is $\sim \big(\delta + (s + \Delta_{\max})L\big)T$. Therefore, the overall regret is $\sim s^{-1}\delta^{-2}\log T + \big(\delta + (s + \Delta_{\max})L\big)T$.

Smooth Reward Functions. In the above proof, we used the fact that if the exploitation price is ε distance away from the optimal price, then an $O(\varepsilon)$ regret is incurred per round. If we assume the second derivative of each reward function exists and is bounded by a constant C, then by Taylor expansion,

$$\begin{aligned} |R(x) - R(p^*)| &= |R'(p^*)(x - p^*) + \frac{1}{2}R''(p^*) \cdot (x - p^*)^2 + o(|x - p^*|^2)| \\ &\leq 0 + \frac{C}{2}|x - p^*|^2 + o(|x - p^*|^2) = \left(\frac{C}{2} + o(1)\right) \cdot |x - p^*|^2, \quad \text{as } |x - p^*| \to 0. \end{aligned}$$

In other words, an ε-error in the estimation of optimal price only incurs regret $O(\varepsilon^2)$ per round. This suggests that the $\tilde{O}(T^{3/4})$ upper bound may be improved under suitable smoothness assumptions.

Definition 3 (Smooth Reward Functions). Given $L, C > 0$, define $\hat{\mathcal{F}}_{L,C}$ to be the family of all unimodal L-Lipschitz twice-differentiable reward functions from $[0,1]$ to $[0,1]$, whose second-derivatives are bounded by a common constant C in absolute value.

It turns out that an improvement can be achieved by simply choosing a different combination of s, δ in the UE policy. In fact, the proof is almost identical to the analysis above except that the term s in Lemma 16 can now be strengthened to $O(s^2)$, as formally stated below.

Lemma 17. *If x_h denotes the halting price of* $\mathrm{UE}_{s,\delta}$ *on a reward function $R \in \hat{\mathcal{F}}_{L,C}$, then conditional on event $\mathcal{C}$, $R(x_h) \geq R(p^*) - (2C + L)s^2 - 12\delta$.*

Proof. As in the proof of Lemma 16, let x_h be the halting price, x_j be the price with highest LCB, and x_ℓ be the lowest price above p^*. The crux is bounding $|R(x_h) - R(x_{h-1})|$: we are able to improve the dependence on s from s to s^2, as stated below.

Claim 1. *$|R(x_h) - R(x_{h-1})| \leq (C + L)s^2 + 4\delta$.*

We complete the proof assuming this lemma. Since $R'(p^*) = 0$, by Taylor expansion

$$|R(p^*) - R(x_\ell)| = \left| R'(p^*) \cdot (p^* - x_\ell) + \frac{1}{2}R''(p^*) \cdot (p^* - x_\ell)^2 + o(|p^* - x_\ell|^2) \right| = \left(\frac{C}{2} + o(1)\right) \cdot s^2, \quad \text{as } s \to 0.$$

In particular, for large T, the $o(1)$ term above becomes less than $\frac{C}{2}$, and

$$|R(p^*) - R(x_\ell)| \le Cs^2. \tag{17}$$

Recall that d_i is the empirical mean demand in the UE policy at x_i for each i. By definition of j, we have $\bar{d}_j \ge \bar{d}_\ell$, so conditional on event $\mathcal{C}$,

$$R(x_j) \ge \bar{x}_j \cdot d_j - 2\delta \ge \bar{x}_\ell \cdot d_\ell - 2\delta \ge R(x_\ell) - 4\delta. \tag{18}$$

Moreover, since the confidence interval at x_{h-1} and x_j are intersecting,

$$|R(x_j) - R(x_{h-1})| \le 4\delta. \tag{19}$$

Combining (17),(18),(19), we obtain

$$R(x_{h-1}) \ge R(p^*) - 8\delta - Cs^2.$$

The proof completes by combining with Claim 1. $\square$

Proof of Claim 1. By the Intermediate Value Theorem, there exists $\xi \in [x_{h-1}, x_{h-2}]$ s.t.

$$sR'(\xi) = R(x_{h-1}) - R(x_{h-2}).$$

Conditional on event $\mathcal{C}$, since the confidence intervals at x_{h-1} and x_{h-2} are intersecting, we have $|R(x_{h-1}) - R(x_{h-2})| \le 4\delta$, hence $sR'(\xi) \le 4\delta$, i.e.

$$R'(\xi) \le \frac{4\delta}{s}.$$

Thus by L-Lipschitzness,

$$R'(x_{h-1}) \le R'(\xi) + sL \le \frac{4\delta}{s} + sL.$$

Moreover, since we have conditioned on event $\mathcal{C}$, it holds that $x_{h-1} \le p^*$, hence $R'(x_{h-1}) \ge 0$ by unimodality of R. By Taylor's Theorem, there exists $\eta \in [x_h, x_{h-1}]$ s.t.

$$\begin{aligned}
|R(x_h) - R(x_{h-1})| &= |R'(x_{h-1})s + R''(\eta)s^2|, \\
&\le |R'(x_{h-1})|s + Cs^2, \\
&\le (\frac{4\delta}{s} + sL) \cdot s + Cs^2 \\
&= (C+L)s^2 + 4\delta. \qquad \square
\end{aligned}$$

Thus, the regret in the exploitation phase becomes $O(s^2 + \delta)T$, so the total regret is now $O\big(s^{-1}\delta^{-2}\log T + (s^2+\delta)T\big)$. Choosing $s \sim T^{-1/7}$ and $\delta \sim T^{-2/7}$, we obtain an $\tilde{O}(T^{5/7})$ regret bound.

Theorem 14 **(Upper Bound for Smooth Reward Functions)**. *For any* $L, C > 0$, *with* $s = (L+C)^{-3/7}T^{-1/7}\log^{1/7}T$ *and* $\delta = (L+C)^{1/7}T^{-2/7}\log^{1/7}T$, *we have*

$$\mathrm{Reg}(\mathrm{UE}_{s,\delta}, \hat{\mathcal{F}}_{L,C}, I = \infty) = O\big((C+L)^{1/7}T^{5/7}\log^{2/7}T\big).$$

III.3.2. Finite Inventory: Proof of Theorem 12 The first hurdle for showing Theorem 12 is that the optimal fixed price no longer enjoys a clean expression. Recall that when $I = \infty$, for any reward function R, the optimal FPP simply selects any $p^* \in \arg\max_{x\in[0,1]} R(x)$. However, this is no longer true when $I < \infty$. In fact, the optimal fixed price p_{OPT} has two equivalent characterizations.

Characterization of the Optimal FPP. It would be more convenient in this section to work with the demand functions (rather than the reward functions). By abuse of notation, write $r(\mathbb{A}, D)$ the expected reward of a policy $\mathbb{A}$ under a demand function D. Similarly define $r(p,D)$ as the expected reward of FPP(p).

Consider the FPP at some price $p \in [0,1]$. Let $\{X_t\}_{t\in[T]}$ be i.i.d. samples at price p from demand function D, which by definition satisfy $D(p) = \mathbb{E}[X_t]$ for all $t \in [T]$.

- Characterization 1: Define the (random) *depletion time* τ_p to be the round when inventory is depleted, i.e. $\tau_p = \min\{t : \sum_{j=1}^{t} X_j \geq I\}$. Then by Wald's identity (see e.g. Mitzenmacher and Upfal (2017)), the reward of FPP(p) is
$$r(p,D) = \mathbb{E}[\sum_{t=1}^{\tau_p} p \cdot X_t] = p \cdot \mathbb{E}[\tau_p] \cdot \mathbb{E}[X_t] = \mathbb{E}[\tau_p] \cdot R(p),$$
where $R(p) = D(p) \cdot p$ is the reward function for D. Thus, $p_{OPT} = \arg\max_{p\in[0,1]}\{\mathbb{E}[\tau_p] \cdot R(p)\}$.
- Characterization 2: Define the (random) *sales* to be $N_p = \min\{I, \sum_{t=1}^{T} X_t\}$. Then, $r(p,D) = \mathbb{E}[N_p] \cdot p$, and hence $p_{OPT} = \arg\max_{p\in[0,1]}\{\mathbb{E}[N_p] \cdot p\}$.

Even though neither characterization leads to a simple precise expression for p_{OPT}, fortunately, we can still find a simple *surrogate optimal price* whose reward well approximates that of the optimal policy. To this aim, we first introduce the *depletion price*, the price at which the inventory is perfectly depleted at the end of the time horizon, if the demands were deterministic.

Definition 4 (Depletion Price). The *depletion price* p_d of a strictly decreasing demand function D is the unique price p such that $D(p) = \mathrm{med}\{D(0), D(1), \rho\} \in (0,1)$.

Definition 5 (Surrogate Optimal Price). The *surrogate optimal price* (SOP) for a demand function D is
$$p_{\mathrm{SOP}} = \max\{p_d, p^*\} = \begin{cases} p^*, & \text{if } p^* \geq p_d, \\ p_d, & \text{if } p^* < p_d. \end{cases}$$
We will first analyze the regret of DUE against the following surrogate regret, and then translate the bound back to the "real" regret using Lemma 3.

Definition 6 (Surrogate Regret). For any policy $\mathbb{A}$ and any demand function D with SOP p_{SOP}, define the *surrogate regret* as $\mathrm{SR}(\mathbb{A}, D) = r(p_{\mathrm{SOP}}, D) - r(\mathbb{A}, D)$. Let $\mathcal{D}$ be any family of demand functions, define $\mathrm{SR}(\mathbb{A}, \mathcal{D}) := \max_{D\in\mathcal{D}} \mathrm{SR}(\mathbb{A}, D)$.

Lemma 18 **(SOP is almost optimal)**. *Let p_d be the depletion price of an L-Lipschitz demand function $D:[0,1]\to[0,1]$. Then for any $\Delta\in(0,\frac{1}{2})$,*

$$r(p_{\mathrm{OPT}},D)-r(p_{\mathrm{SOP}},D)\le\begin{cases}(\Delta+e^{-\Omega(\Delta^2 I)})\cdot T, & \text{if } p^*\ge p_d\\ (\Delta+e^{-\Omega(\Delta^2 I)})\cdot I, & \text{if } p^*<p_d.\end{cases}$$

Proof. We first illustrate the idea for establishing Lemma 18. Our goal is showing that p_{SOP} is nearly optimal. In the first case, $p^*\ge p_d$, we have $p_{\mathrm{SOP}}=p^*$. Since the policy FPP(p^*) is unlikely to deplete the inventory, the problem almost reduces to the infinite inventory version. So by Lipschitzness, $p_{OPT}\approx p^*=p_{SOP}$ and hence p_{SOP} is nearly optimal.

Now suppose $p^*<p_d$, in which case $p_{SOP}=p_d$. Our goal is showing p_d is almost optimal. Note that we do not know whether p_{OPT} is greater or less than p_d, so we need to argue that p_d is both nearly-optimal among prices p lower and higher than p_d, as discussed below.

- Consider $p\le p_d$. Since FPP(p) is likely to sell out all inventory, we have $r(p)\sim pI$. Applying this observation on p_d, we have $r(p,D)\sim pI\le p_dI$ for $p\le p_d$.
- Consider $p\ge p_d$. Since the inventory is unlikely to be depleted by FPP(p), we have $r(p,D)\sim pD(p)T=R(p)T$. By unimodality, R is non-increasing on $[p_d,1]$, so $R(p_d)\ge R(p)$ and hence $r(p_d,D)\sim R(p_d)T>R(p)T\sim r(p,D)$. Thus p_d is almost optimal in $[p_d,1]$.

We now convert the above idea into a formal proof.

Part I. Suppose $p^*\ge p_d$. To lower bound the reward of FPP(p^*), we only need to lower bound its expected stopping time $\mathbb{E}[\tau_{p^*}]$. To this aim, write $\tau=\tau_{p^*}$ and consider $m:=(1-\Delta)T$ i.i.d. demand samples $Z_1,...,Z_m$ at price p^*. Consider $Z:=\sum_{i=1}^m Z_i$. Since $p_d\le p^*$, we have $\mathbb{E}Z_1=D(p^*)\le D(p_d)$, so

$$\mathbb{E}Z=m\cdot\mathbb{E}Z_1\le(1-\Delta)T\cdot D(p_d)=(1-\Delta)I,$$

and hence $I\ge(1-\Delta)^{-1}\mathbb{E}Z\ge(1+\Delta)\mathbb{E}Z$. Hence,

$$\mathbb{P}[\tau<(1-\Delta)T]=\mathbb{P}[Z>I]\ \le\mathbb{P}[Z>(1+\Delta)\mathbb{E}Z]\le e^{-\frac{\Delta^2}{2}\mathbb{E}Z}\le e^{-\Omega(\Delta^2T)},$$

where the last inequality follows from Chernoff bound (Lemma 14), and that $\mathbb{E}Z\ge(1-\Delta)T\cdot D(p_d)\ge\Omega(T)$ for $\Delta\le\frac{1}{2}$. Thus,

$$\begin{aligned}r(p^*,D)&=R(p^*)\cdot\mathbb{E}[\tau]\\&\ge R(p^*)\cdot\mathbb{E}[\tau|\tau\ge(1-\Delta)T]\cdot\mathbb{P}[\tau\ge(1-\Delta)T]\\&\ge R(p^*)\cdot(1-\Delta)T\cdot(1-e^{-\Omega(\Delta^2T)}).\end{aligned}$$

Therefore, for any $p \in [0,1]$,

$$\begin{aligned} r(p,D) - r(p^*,D) &\le R(p)T - R(p^*) \cdot (1-\Delta)T \cdot (1 - e^{-\Omega(\Delta^2 T)}) \\ &\le R(p)T - R(p^*) \cdot (1 - \Delta - e^{-\Omega(\Delta^2 T)})T \\ &= (R(p) - R(p^*))T + R(p^*) \cdot \big(\Delta + e^{-\Omega(\Delta^2 T)}\big)T \\ &\le \big(\Delta + e^{-\Omega(\Delta^2 T)}\big)T, \end{aligned}$$

the last inequality follows since $R(p) \le R(p^*) \le 1$ by definition of p^*.

Part II. Now suppose $p^* \le p_d$. Recall that for any $p \in [0,1]$, N_p, τ_p are the sales and depletion time, and the reward of an FPP can be written as $r(p,D) = \mathbb{E}N_p \cdot p = \mathbb{E}[\tau_p] \cdot R(p)$. We lower bound $r(p_d, D)$ by analyzing both N_{p_d} and τ_{p_d}.

Claim 2. *For any* $\Delta > 0$, $\mathbb{P}[N_{p_d} < (1-\Delta)I] \le e^{-\Omega(\Delta^2 I)}$, *and* $\mathbb{P}[\tau_{p_d} < (1-\Delta)T] \le e^{-\Omega(\Delta^2 I)}$.

We defer the proofs of the claim to the end. Then for any $p \le p_d$,

$$\begin{aligned} r(p,R) - r(p_{SOP},R) &= r(p,R) - r(p_d,R) \\ &\le pI - p_d \cdot \mathbb{E}[N_{p_d}] \\ &\le pI - p_d \cdot \mathbb{E}[N_{p_d} | N_{p_d} \ge (1-\Delta)I] \cdot \mathbb{P}[N_{p_d} \ge (1-\Delta)I] \\ &\le pI - p_d \cdot (1 - e^{-\Omega(\Delta^2 I)}) \cdot (1-\Delta)I && \text{By Claim 2} \\ &\le pI - p_d \cdot I \cdot (1 - \Delta - e^{-\Omega(\Delta^2 T)}) \\ &= (p - p_d)I + p_d \cdot (\Delta + e^{-\Omega(\Delta^2 I)})I \\ &\le (\Delta + e^{-\Omega(\Delta^2 I)})I. && \text{Since } p_d \le 1 \end{aligned}$$

On the other hand, for any $p > p_d$,

$$\begin{aligned} r(p,R) - r(p_{SOP},R) &= r(p,R) - r(p_d,R) \\ &\le R(p) \cdot T - R(p_d) \cdot \mathbb{E}[\tau_{p_d}] \\ &\le R(p) \cdot T - R(p_d) \cdot \mathbb{E}[\tau_{p_d} | \tau \ge (1-\Delta)T] \cdot \mathbb{P}[\tau \ge (1-\Delta)T] \\ &\le R(p) \cdot T - R(p_d) \cdot (1-\Delta)T \cdot (1 - e^{-\Omega(\Delta^2 I)}) && \text{By Claim 2} \\ &\le (R(p) - R(p_d))T + (\Delta + e^{-\Omega(\Delta^2 I)})T. \end{aligned}$$

Note that $p \ge p_d \ge p^*$ and R is unimodal with maximum attained at p^*, so $R(p) \le R(p_d)$. Therefore,

$$r(p,R) - r(p_{SOP},R) \le (\Delta + e^{-\Omega(\Delta^2 I)})T. \qquad \square$$

Proof of Claim 2. We first bound $\mathbb{P}[N_{p_d} < (1-\Delta)I]$. Let $X_1, ..., X_T$ be i.i.d. drawn demands at p_d and write $X = \sum_{t=1}^{T} X_t$. Note that $X < (1-\Delta)I$ if and only if $N_{p_d} < (1-\Delta)I$. Since $\mathbb{E}X = I$ and, by Hoeffding inequality (Lemma 14),

$$\mathbb{P}[N_{p_d} < (1-\Delta)I] = \mathbb{P}[X < (1-\Delta)I] = \mathbb{P}[X < (1-\Delta)\mathbb{E}X] \leq e^{-\frac{1}{2}\Delta^2 I}.$$

Next we bound $\mathbb{P}[\tau_{p_d} < (1-\Delta)T]$. Let $Z_1, ..., Z_m$ be i.i.d. drawn demands at price p_d where $m = (1-\Delta)T$. Then,

$$\begin{aligned}
\mathbb{P}[\tau_{p_d} < (1-\Delta)T] &= \mathbb{P}[Z \geq I] \\
&= \mathbb{P}[Z \geq \frac{1}{1-\Delta}\mathbb{E}Z] && \text{Since } \mathbb{E}Z = (1-\Delta)I \\
&\leq \mathbb{P}[Z \geq (1+\Delta)\mathbb{E}Z] \\
&\leq e^{-\frac{\Delta^2}{2}\mathbb{E}Z} && \text{By Chernoff bound (Lemma 14)} \\
&\leq e^{-\Omega(\Delta^2 I)}. && \square
\end{aligned}$$

Since we assumed $I = \Omega(T)$, the regret of p_{SOP} in Lemma 18 can be simplified to $O(\Delta T + e^{-\Omega(\Delta^2 T)}T)$, which becomes $O(\sqrt{T}\log^{1/2} T)$ if we select $\Delta = T^{-1/2}\log^{1/2} T$, and we obtain the following.

Proposition 3. *Let $\mathbb{A}$ be any markdown policy, then for any L-Lipschitz demand function D,*

$$\text{Reg}(\mathbb{A}, D) \leq \text{SR}(\mathbb{A}, D) + O(\sqrt{T}\log^{1/2} T).$$

As in Section III.3, we denote $x_j = 1 - js$ the j-th sample price for $j = 1, ..., s^{-1}$, and $\bar{d}_j$ the mean demand at x_j as defined in Step 5 of Algorithm 7.

We first state and prove some useful lemmas. Recall that $x_j = 1 - js$ and $\bar{d}_j$ is the empirical mean demand at x_j over $O(\delta^{-2}\log T)$ samples as described in policy $\text{DUE}_{s,\delta}$. The following can be obtained immediately by combining the Hoeffding bound (Lemma 14) and union bound.

Lemma 19. *Let $\mathcal{C}$ be the event that $|\bar{d}_j - D(p)| \leq \delta$ for all sample prices x_j. Then $\mathbb{P}[\bar{\mathcal{C}}] \leq O(T^{-2})$.*

By definition of Algorithm 7, the exploration phase of the DUE policy must terminate due to one of the following three events:

- E_0: Inventory runs out.
- E_1: $UCB(x) < LCB_{max}$ where x is the current sample price.
- E_2: $(\bar{d} + \delta) \cdot T \geq I$ where $\bar{d}$ is the empirical mean of the current sample price.

Recall that the halting price x_h is the price where the policy finds sufficient evidence that the price has dropped below either p_d or p^*, hence stops exploration. We next show that the halting price is unlikely to be too much lower than p_d.

Lemma 20. *If event $\mathcal{C}$ occurs, then $x_h \geq p_d - s$.*

Proof. Suppose $\mathcal{C}$ occurs. Consider $k = \min\{j : x_j < p_d\}$, so that x_k is the highest sample price less than p_d. By definition of $\mathcal{C}$, the empirical mean demand $\bar{d}_k$ at x_k satisfies $\bar{d}_k + \delta \geq D(x_k)$. By monotonicity of the demand function, we have $D(x_k) \cdot T \geq D(p_d) \cdot T = I$, thus the halting condition would be satisfied at x_k, if not earlier. Therefore, $x_h \geq x_k \geq p_d - s$. □

We next show that E_0 is unlikely to occur during the exploration phase.

Lemma 21. *Suppose $\delta^{-2}s^{-1} = O(T^{0.99})$, then $\mathbb{P}[E_0|\mathcal{C}] \leq T^{-2}$.*

Proof. For any sample price x, let $Z_1(x), ..., Z_m(x)$ be i.i.d. samples from the demand distribution at price x where $m = \lceil 3\delta^{-2}\log T \rceil$. By the above lemma, conditional on $\mathcal{C}$, we have $x_h \geq x_k \geq p_d - s$. Thus, if E_0 occurs during the exploration phase, then $\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I$, hence the problem reduces to showing $\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I|\mathcal{C}] \leq T^{-2}$. We bound this probability as follows.

$$\begin{aligned}
\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I|\mathcal{C}] &\leq \frac{\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I]}{\mathbb{P}[\mathcal{C}]} \\
&= \frac{\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I]}{1 - \mathbb{P}[\overline{\mathcal{C}}]} \\
&\leq 2\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I],
\end{aligned}$$

where the last inequality follows since $(1-\varepsilon)^{-1} \leq 2$ for any $\varepsilon \in (0, \frac{1}{2})$. We complete the proof by bounding $\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I]$. Note that for any $x \geq p_d$, we have $\mathbb{E}[Z_j(x)] = D(x)$ for $j \in [m]$. Thus,

$$\begin{aligned}
\mathbb{E}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i)] &= \sum_{i=1}^k D(x_i)m \\
&\leq km \cdot (D(p_d) + sL) \\
&\leq 3s^{-1}\delta^{-2}\log T \cdot (\frac{I}{T} + sL) \\
&\leq O\big((T^{0.99} + L) \cdot \log T \cdot T^{-1}\big) \cdot I = o(I) \qquad \text{Since } \delta^{-2}s^{-1} = O(T^{0.99})
\end{aligned}$$

Applying Hoeffding bound (Lemma 14), we immediately obtain that $\mathbb{P}[\sum_{i=1}^k \sum_{j=1}^m Z_j(x_i) > I] \leq T^{-3}$, and the proof follows. □

The proof of Lemma 3 can be split into two parts, stated in Lemmas 22 and 24. Following the ideas of Lemma 13, we may bound the regret when $p^* \geq p_d$.

Lemma 22. *Suppose $p^* \geq p_d$. Then for any $\delta, s \in (0,1)$ such that $\delta^{-2}s^{-1} = O(T^{0.99})$,*

$$\mathrm{SR}(\mathrm{DUE}_{s,\delta}, D) = O\big(s^{-1}\delta^{-2}\log T + (sL+\delta)T\big).$$

Proof. We condition on events $\mathcal{C}$ and $\bar{E}_0$. The proof is similar to Proposition 13. Recall $R^* = \max_{x\in[0,1]} R(x)$ and x_k is the maximum sample price not exceeding p_d. There are two cases.

- If E_1 occurs at some sample price $x_h > x_k$, then by the same argument in Proposition 13, $R(x_h) \geq R^* - O(sL+\delta)$.
- Otherwise, E_1 does not occur at any sample price $x_j > x_k$, then E_2 occurs and $x_h = x_k$. In both cases, $UCB(x_{h-1}) \geq LCB_{max}$, and by mimicking the proof of Proposition 13, we deduce that $R(x_k) \geq R^* - O(sL+\delta)$.

Thus in either case, the regret in the exploitation phase is $O\big((\delta + sL)T\big)$. Since there are $O(s^{-1})$ sampling prices and we selected each for $O(\delta^{-2}\log T)$ times, the regret in the exploration phase is $O(\delta^{-2}s^{-1}\log T)$, thus the regret conditional on $\mathcal{C}$ and $\bar{E}_0$ is $O(\delta^{-2}s^{-1}\log T + (sL+\delta)T)$. Finally,

$$\begin{aligned}\mathrm{SR}(\mathrm{DUE}_{s,\delta}, D) &\leq \mathbb{P}[\mathcal{C} \wedge \bar{E}_0]\cdot O(\delta^{-2}s^{-1}\log T + (sL+\delta)T) + \mathbb{P}[E_0]\cdot T + \mathbb{P}[\bar{\mathcal{C}}]\cdot T \\ &\leq O(\delta^{-2}s^{-1}\log T + (sL+\delta)T). \qquad \square\end{aligned}$$

When $p^* \leq p_d$ the analysis becomes more involved. In Lemma 20 we showed that the DUE policy is unlikely to halt "too late", i.e. x_h is not too much lower than p_d. To show Lemma 24, we next rule out the other unfavorable event, that the policy halts "too early".

Lemma 23. *Suppose $p^* \leq p_d$ and condition on event $\mathcal{C}$, it holds $D(x_h) \geq D(p_d) - 2\delta$.*

Proof. We start with the trivial case $x_h < p_d$: in this case, by monotonicity of demand functions, $D(x_h) \geq D(p_d)$ and the claim holds. Now suppose $x_h \geq p_d$. By unimodalty, R is non-increasing on $[p^*, 1]$, thus event E_1 won't occur at any sample price $x_j \geq p^*$. In other words, the exploration phase must have terminated due to E_2, so $(\bar{d}(x_h) + \delta)\cdot T \geq I$. Since we have conditioned on $\mathcal{C}$, it follows that $|\bar{d}(x_h) - D(x_h)| \leq \delta$. Therefore, $D(x_h) \geq \bar{d}(x_h) - \delta \geq \frac{I}{T} - 2\delta = D(p_d) - 2\delta$, and the proof follows. □

Before presenting the formal proof Lemma 24, we first expose the technical challenge. To upper bound the surrogate regret, it suffices to lower bound the expected reward of $\mathrm{DUE}_{s,\delta}$, which reduces to lower-bounding $\mathbb{E}N$ where N is the sales of the policy. However, since DUE is no longer an FPP, we can not imitate the proof of Lemma 18. Fortunately, the exploitation phase can be viewed as an FPP at the halting price x_h. This, however, does not lead to a straightforward analysis, since the number of rounds and inventory level in the exploitation phase are *random*, determined by the realizations in the exploration phase.

To circumvent this issue, we perform a conservative analysis on the exploitation phase reward. Since the number of samples at each round is $3\delta^{-2}\log T$ and there are at most s^{-1} sample prices, it would take at most $3s^{-1}\delta^{-2}\log T$ rounds to enter the exploitation phase. In this case, the exploration phase has $T' := T - 3s^{-1}\delta^{-2}\log T$ rounds, ending with inventory at least $I' := I - 3s^{-1}\delta^{-2}\log T$. Thus, it suffices to focus on lower bounding the reward of the FPP at x_h that starts with I' inventory and lasts for T' rounds. We formalize this idea below.

LEMMA 24. *When $p^* \le p_d$, then for any $s,\delta,\varepsilon > 0$ with $\varepsilon \ge 3(1-\rho)s^{-1}\delta^{-2}I^{-1}\log T + sL\rho^{-1}$,*

$$\mathrm{SR}(\mathrm{DUE}_{s,\delta}, D) = O(\delta T + sI + e^{-\Omega(\varepsilon^2 T)}I + \varepsilon I + s^{-1}\delta^{-2}\log T).$$

Proof. Let $Z_1, ...Z_{T'}$ be i.i.d. samples drawn from the demand distribution at price x_h and $Z = \sum_{t=1}^{T'} Z_t$. We will focus on lower bounding the expectation of $N' := \min\{Z, I'\}$, i.e. the sales in with T' rounds and initial inventory I'. Since

$$\mathbb{E}N' \ge \mathbb{P}[N' > (1-\varepsilon)\mathbb{E}Z] \cdot \mathbb{E}[N' | N' > (1-\varepsilon)\mathbb{E}Z], \tag{20}$$

and our goal becomes lower bounding $\mathbb{P}[N' > (1-\varepsilon)\mathbb{E}Z]$, or, upper bounding $\mathbb{P}[N' \le (1-\varepsilon)\mathbb{E}Z]$.

The hurdle for bounding the above is that $N' = \min\{Z, I'\}$ is a *truncated* sum of Z_t's instead of an i.i.d. sum. Fortunately, we observe that if $I' < (1-\varepsilon)\mathbb{E}Z$, then the event $\{N' \le (1-\varepsilon)\mathbb{E}Z\}$ is equivalent to $\{Z \le (1-\varepsilon)\mathbb{E}Z\}$, thereby we may proceed with concentration bounds on Z. We now derive a sufficient condition for having $I' < (1-\varepsilon)\mathbb{E}Z$. Recall that $\rho = I/T < 1$.

CLAIM 3. *Let $\alpha = 3I^{-1}s^{-1}\delta^{-2}\log T$, i.e. the maximum possible proportion of inventory consumed in the first $I' = 3s^{-1}\delta^{-2}\log T$ rounds. Then for any ε satisfying $\varepsilon \ge (1-\rho)\alpha + sL\rho^{-1}$, it holds that $(1-\varepsilon)\mathbb{E}Z \le I'$.*

Proof.

$$\begin{aligned}(1-\varepsilon)\mathbb{E}Z \le I' &\iff (1-\varepsilon)\cdot T' \cdot D(x_h) \le I' \\ &\iff 1-\varepsilon \le \frac{I'}{T'D(x_h)} \\ &\Longleftarrow 1-\varepsilon \le \frac{I'}{T'(D(p_d)+sL)} && \text{By Lemma 20}\end{aligned}$$

$$
\begin{aligned}
&\iff 1-\varepsilon = \frac{I'T}{T'(I+sLT)} && \text{Multiply numerator and denominator by } T\\
&\iff 1-\varepsilon = \frac{(1-\alpha)\cdot I\cdot T}{(1-\rho\alpha)\cdot T\cdot (I+sLT)}\\
&\iff 1-\varepsilon = \frac{1-\alpha}{1-\rho\alpha}\cdot\frac{1}{1+sL\rho^{-1}}\\
&\impliedby 1-\varepsilon \le (1-\alpha)\cdot(1+\rho\alpha)\cdot(1-sL\rho^{-1}) && \text{Since } (1-z)^{-1}\ge 1+z, \forall z<1\\
&\impliedby 1-\varepsilon \le (1-(1-\rho)\alpha)\cdot(1-sL\rho^{-1})\\
&\impliedby \varepsilon \ge (1-\rho)\alpha + sL\rho^{-1}. && \square
\end{aligned}
$$

Thus, for any $\varepsilon \ge (1-\rho)\alpha + sL\rho^{-1}$ it holds that

$$\mathbb{P}[N' \le (1-\varepsilon)\mathbb{E}Z] = \mathbb{P}[Z \le (1-\varepsilon)\mathbb{E}Z] \le e^{-\frac{\varepsilon^2}{2}\mathbb{E}Z} \le e^{-\Omega(\varepsilon^2 T)}. \tag{21}$$

By Lemma 23 and the definition of Z, we have

$$
\begin{aligned}
\mathbb{E}Z = D(x_h)\cdot T' &\ge (D(p_d)-2\delta)\cdot(1-\alpha\rho)T\\
&\ge (D(p_d)-2\delta-\alpha\rho)\cdot T\\
&= I-(2\delta+\alpha\rho)T,
\end{aligned}\tag{22}
$$

where the last step follows since $D(p_d)\cdot T = I$. Combining (20), (21) and (22), we are able to lower bound the reward of the DUE policy as

$$
\begin{aligned}
r(\mathrm{DUE}_{s,\delta}, D) &\ge x_h\cdot\mathbb{E}[N']\\
&\ge (p_d-s)\cdot\mathbb{P}[N'>(1-\varepsilon)\mathbb{E}Z]\cdot\mathbb{E}[N'|N'>(1-\varepsilon)\mathbb{E}Z]\\
&\ge (p_d-s)\cdot(1-e^{-\Omega(\varepsilon^2T)})\cdot(1-\varepsilon)\cdot\mathbb{E}Z\\
&\ge (p_d-s)\cdot(1-e^{-\Omega(\varepsilon^2T)})\cdot(1-\varepsilon)\cdot(I-(2\delta+\alpha\rho)T)\\
&\ge (p_dI-sI-2\delta T-\alpha I)\cdot(1-e^{-\Omega(\varepsilon^2T)}-\varepsilon)\\
&\ge p_dI-sI-2\delta T-\alpha I-e^{-\Omega(\varepsilon^2T)}I-\varepsilon I.
\end{aligned}
$$

Hence, recalling that $\alpha I = 3s^{-1}\delta^{-2}\log T$, we have

$$
\begin{aligned}
r(p_{SOP}, D) - r(\mathbb{A}, D) &= r(p_d, D) - r(\mathbb{A}, D)\\
&\le p_dI - (p_dI - 2\delta T - sI - e^{-\Omega(\varepsilon^2T)}I - \varepsilon I - \alpha I)\\
&\le 2\delta T + sI + e^{-\Omega(\varepsilon^2T)}I + \varepsilon I + 3s^{-1}\delta^{-2}\log T. \qquad \square
\end{aligned}
$$

Theorem 12 immediately follows by combining the Lemma 22, Lemma 24 and Corollary 3, with $\delta = T^{-1/4}(L\log T)^{1/4}, s=\delta/L$ and $\varepsilon = 3(1-\rho)s^{-1}\delta^{-2}I^{-1}\log T + sL\rho^{-1}$.

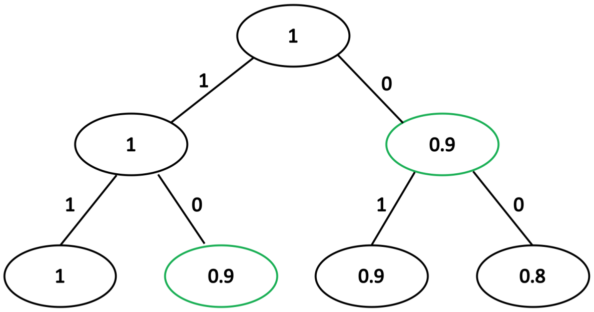

Figure 2 Viewing a policy as a decision tree. Entrance nodes of $[0.89, 0.91]$ are drawn in green.

III.4. Proof of Lower Bound (Theorem 13)

We now turn to proving our lower bound, which establishes minimax optimality of the policy described in the previous section in the setting of infinite inventory. Without loss of generality (by re-scaling) we assume $\rho = \frac{I}{T} = 1$, and thus abbreviate $\text{Reg}(\mathbb{A}, \mathcal{M}, I)$ as $\text{Reg}(\mathbb{A}, \mathcal{M})$ for simplicity.

III.4.1. Preliminaries Our proof considers *Bernoulli* reward distribution at each price and employs the following alternate view of a *policy* as binary decision trees (see Fig 2), which we will make precise in this section.

Definition 7 (Prefix). Let $\{0,1\}^* = \bigcup_{n=1}^{\infty} \{0,1\}^n \cup \{\text{null}\}$ be the set of all finite-length binary vectors, where *null* denotes the empty binary vector. For any $v \in \{0,1\}^*$ and $k \in \mathbb{Z}$, the *k-prefix* of v is defined as $v^k = (v_1, ..., v_k)$.

We will consider probability spaces on sets containing the prefixes of every element.

Definition 8 (Downward Closed Set). For any $v, w \in \{0,1\}^*$, we define $w \prec v$ if there exists $k \in \mathbb{Z}$ such that $v^k = w$. A set $\Omega \in \{0,1\}^*$ is *downward closed*, if for any $v \in \Omega$ and $w \prec v$, we have $w \in \Omega$.

A decision tree is specified by a downward closed set equipped with a real-valued function.

Definition 9 (Decision Tree). A *binary decision tree* is a tuple (Ω, x) where $\Omega \subseteq \{0,1\}^*$ is downward closed and $x : \Omega \to \mathbb{R}$ is a mapping. Moreover, each $v \in \Omega$ is called a *node*.

Intuitively, for each node $v = (v_1, ..., v_k)$, the value $x(v)$ is just the *price* that the policy selects upon observing demands $v_1, ..., v_k$ at prices $x(v^1), ..., x(v^k)$. Recalling that we have normalized the price space to be $[0,1]$, so we will subsequently consider only decision trees (Ω, x) with $0 \le x(v) \le 1$ for all $v \in \Omega$. For notational convenience, we suppress the notation $x(v)$ simply as x_v.

We next introduce an equivalent definition of a markdown policy, using the language of decision trees.

Definition 10 (Markdown Policy, Equivalent Definition). A *markdown policy* is a decision tree (Ω, x) such that $x(v^1) \geq x(v^2) \geq ... \geq x(v^k)$ for any $v = (v_1, ..., v_k) \in \Omega$.

One may verify that this definition of markdown policy is indeed equivalent with the one given in Section III.2. We next introduce some standard terminologies for decision trees, in case the reader is not familiar with graph theory.

Definition 11 (Decision Tree Basics). Let $\mathbb{A} = (\Omega, x)$ be a decision tree and $v, w \in \Omega$.

i). We say v is a *leaf* if there does not exist $w \in \Omega$ with $v \prec w$.

ii). The *depth* $d(v)$ of v is defined to be the length of binary vector v. Denote $L(\Omega) \subseteq \Omega$ the subset of all leaves. Each node in $\Omega \backslash L(\Omega)$ is called an *internal* node.

iii). We say w is an *ancestor* of v if $w \prec v$. If in addition, $d(v) = d(w) + 1$, then we say w is the *parent* of v and denote $w = par(v)$, and say v is a *child* of w.

iv). A decision tree is *binary* if every internal has exactly two children.

Given a binary decision tree, every reward function induces a natural probability measure over the leaves. In fact, consider a random walk from the root to a random leaf, where at each internal node v, the walk moves to one of the two children with probability $R(x_v)$ and $1 - R(x_v)$ respectively. We formally define this probability measure below.

Definition 12 (Probability Measure on Leaves). Let (Ω, x) be a decision tree and $R : [0,1] \to [0,1]$. Write $L = L(\Omega)$. For each $\ell = (\ell_1, ..., \ell_d) \in L$, define

$$p_R(\ell) = \prod_{j=1}^{d} R\left(x(\ell^j)\right)^{\ell_j} \cdot \left(1 - R\left(x(\ell^j)\right)\right)^{1-\ell_j}$$

The probability measure $\mathbb{P}_R$ on $(\Omega, 2^L)$ is then given by $\mathbb{P}_R(S) = \sum_{\ell \in S} p_R(S)$ for each $S \subseteq L$. We also define $\mathbb{E}_R$ to be the expectation under the probability measure $\mathbb{P}_R$.

The following lemma argues for the proof of our lower bound, we may restrict our attention to policies in which the prices never change in the second half of $[T]$.

Lemma 25. *Given any markdown policy* $\mathbb{A} = (\{0,1\}^T, x')$*, there is another markdown policy* $\mathbb{B} = (\{0,1\}^T, x)$ *such that*

i. for any $\ell \in L(\{0,1\}^T)$*, we have* $x(\ell^t) = ... = x(\ell^T)$ *for any* $t = \lceil \frac{T}{2} \rceil, ..., T$*, and*

ii. for all $R \in \hat{\mathcal{F}}_L$*,* $\text{Reg}(\mathbb{B}, R) \leq 2 \cdot \text{Reg}(\mathbb{A}, R)$*.*

We defer the proof to the end of this section. In our analysis of lower bound, we will examine the number of rounds that the policy selects a price in an interval. To this aim, we formalize next what it means to “enter” an interval.

Definition 13 (Entrance). Given an interval $[a,b]$ and a decision tree (Ω,x), we say $v\in\Omega$ is an $[a,b]$-*entrance* (or simply, *entrance*), if $x_v \le b$ and $x_{par(v)} > b$. If $\ell\in L(\Omega)$ has no ancestor which is also an entrance, then define $\ell^{\lceil T/2\rceil}$ to be an entrance.

One can easily verify that due to the markdown constraint, there exists exactly one $[a,b]$-entrance on the path from the root to every leaf.

III.4.2. Wald-Wolfowitz Theorem and the Proof of Our Lower Bound Our proof relies on sample complexity lower bound for distinguishing between two distributions, formally defined as follows.

Definition 14 (Adaptive Classifier). Consider $R,B:[0,1]\to[0,1]$. Let (Ω,x) be a decision tree and $f:L(\Omega)\to\{R,B\}$. Then, (Ω,x,f) is called an *adaptive classifier* for R and B. Moreover, given constants $\alpha,\beta\in[0,1]$, an adaptive-classifier (Ω,x,f) is called (α,β)-*confident* if

$$\begin{aligned} \mathbb{P}_R\left(f^{-1}(R)\right) &\ge \alpha, \quad (\textbf{D}\text{etection probability is high})\\ \text{and } \mathbb{P}_B\left(f^{-1}(R)\right) &\le \beta. \quad (\textbf{F}\text{alse-}\textbf{A}\text{larm probability is low})\end{aligned}$$

Our lower bound results all rely upon a Theorem due to Wald and Wolfowitz (1948) for adaptive sequential hypothesis testing, which states that the *expected* number of samples collected in order to **adaptively** distinguish between a pair of distributions R,B must be lower bounded by a function of α,β and the KL-divergence.

Theorem 15 **(Wald-Wolfowitz Theorem)**. *Consider $R,B:[0,1]\to[0,1]$ and an (α,β)-confident adaptive classifier (Ω,x,f). Denote $\Delta(R,B)=\max_{v\in\Omega}\mathrm{KL}\big(R(x_v),B(x_v)\big)$. Let $D(\ell)$ be the depth of leaf $\ell\in L(\Omega)$. Then,*

$$\mathbb{E}_R[D]\ge\frac{\alpha\log\frac{\alpha}{\beta}+(1-\alpha)\log\frac{1-\alpha}{1-\beta}}{\Delta(R,B)},\quad \textit{and}\quad \mathbb{E}_B[D]\ge\frac{\beta\log\frac{\beta}{\alpha}+(1-\beta)\log\frac{1-\beta}{1-\alpha}}{\Delta(B,R)}. \tag{23}$$

Our proof for Theorem 13 considers the following family of reward curves (see Fig 3). Each curve has slope L and -1 on the left and right of its unique optimal price p, and truncated from below to ensure non-negativity[***]. Formally, for $p,x\in[0,1]$, consider

$$R_p(x)=\begin{cases}1-x, & \text{if } x\ge p,\\ 1-(L+1)p+Lx, & \text{if } \frac{(L+1)p-1}{L}\le x\le p,\\ 0, & \text{otherwise,}\end{cases}$$

and $\mathcal{M}=\{R_p : p\in[\frac{1}{8},\frac{7}{8}]\}$.

[***] We assumed that $L\ge 1$.

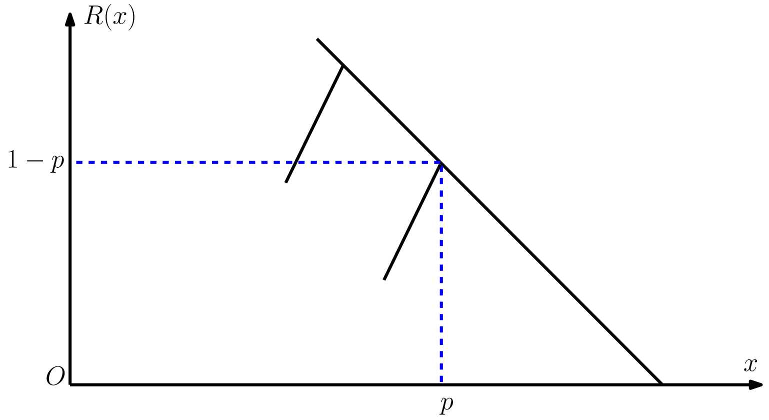

Figure 3 **Instances for showing the lower bound.**

At a high level, we show that any reasonable policy must be able to distinguish between $R = R_p$ with $p = \frac{1}{8}$ and $B = R_b$ for any $b > \frac{1}{2}$. As a result, the price must stay close to b for many rounds, leading to high regret if R is the true reward function.

The technical crux of the proof lies in formalizing the notion of "distinguish between". To this aim, we introduce an *induced* classifier, based on whether it predominantly uses prices in the range $[a,b]$ or below a. Finally, we conclude the proof by showing the expected depth of the induced classifier-tree is $\Omega(T^{\frac{1}{2}})$. We next formally define induced classifier and the leaf coloring.

Definition 15 ((v,a,b)-Classifier). Let (Ω, x) be a decision tree and $[a,b]$ be some interval, and consider an $[a,b]$-entrance v. The (v,a,b)-*classifier* is specified by a tuple $(\tilde{\Omega}, \tilde{x}, \tilde{f})$, where

$$\tilde{\Omega} = \{w \in \Omega : w^{d(v)} = v^{d(v)},\ d(w) \le d(v) + \left\lceil \frac{T}{4} \right\rceil, \text{ and } x(w) \ge a\},$$

$\tilde{x} = x|_{\tilde{\Omega}}$, and the *leaf-coloring* $\tilde{f}$ is given by

$$\begin{aligned} \tilde{f} : L(\tilde{\Omega}) &\to \{R, B\} \\ \ell &\mapsto \begin{cases} R, & \text{if} \quad d(\ell) - d(v) < \lceil \frac{T}{4} \rceil, \\ B, & \text{if} \quad d(\ell) - d(v) = \lceil \frac{T}{4} \rceil. \end{cases} \end{aligned}$$

In words, given a decision tree and an $[a,b]$-entrance v, the (v,a,b)-classifier is obtained by the following procedure:

1. (Truncation) Remove all nodes that are at least $\frac{T}{4}$ levels below v. If a descendent u of v has node-price $x_u \le a$, then remove all descendants of u, hence u becomes a leaf. The subtree rooted at the entrance v after carrying out these descendant removals and the truncation in the previous step is denoted $\mathbb{T}_v$.
2. (Coloring) For every leaf $\ell \in \mathbb{T}_v$, if $x_\ell \le a$, set $f(\ell) = R$, else $f(\ell) = B$.

We next formalize the notion of a tree differentiating between reward functions R or B using the induced probability measure on the leaves of the tree.

Definition 16 (Confidence of (v,a,b)-Tree). A (v,a,b)-classifier $(\tilde{\Omega},\tilde{x},\tilde{f})$ is said to be *confident* if $(\tilde{\Omega},\tilde{x},\tilde{f})$ is $(\frac{2}{3},\frac{1}{3})$-confident[†††] for distinguishing between $R_{\frac{1}{8}}$ and R_b. Moreover, an entrance v is said to be *confident*, if the (v,a,b)-classifier is confident.

We next introduce a key quantity. Let $\mathbb{A}=(\Omega,x)$ be a decision tree and $0\le a\le b\le 1$. The random variable $N=N(\mathbb{A};a,b)$ is defined as

$$\begin{aligned} N : L(\Omega) &\to \mathbb{R} \\ \ell &\mapsto \sum_{i=1}^{d(\ell)} \mathbb{1}\left[x(\ell^i)\in[a,b]\right] \end{aligned}$$

Intuitively, $N(\mathbb{A};a,b)$ is simply the number of times that policy $\mathbb{A}$ selects a price in $[a,b]$. We next present the key lemma concerning $N(\mathbb{A};a,b)$ for any low-regret policy $\mathbb{A}$.

Lemma 26 **(Key Lemma)**. *Let $\mathbb{A}$ be a markdown policy with* $\mathrm{Reg}(\mathbb{A},\mathcal{M})\le\frac{1}{48}L^{1/4}T^{3/4}$ *for all $T>2^{12}L$. If $[a,b]\subset[\frac{3}{4},\frac{7}{8}]$ where $b=a+L^{-3/4}T^{-1/4}$ and $p=\frac{1}{8}$, then*

$$\mathbb{E}_{R_p}\left[N(\mathbb{A};a,b)\right]=\Omega(L^{-1/2}T^{1/2}).$$

We will also use the following folklore regret-decomposition lemma (see e.g. Lemma 4.5 of Lattimore and Szepesvári (2020)).

Lemma 27 **(Decomposition of Regret)**. *Consider a Multi-armed Bandit instance I where each arm $i\in[K]$ has mean reward $\mu_i\in[0,1]$. Let $\mu^*=\max_{i\in[K]}\mu_i$ and N_i be the number of times arm i is selected by a bandit policy $\mathbb{A}$. Then,*

$$Reg(\mathbb{A},I)=\sum_{i\in[K]}(\mu^*-\mu_i)\cdot\mathbb{E}N_i.$$

As remarked in Lattimore and Szepesvári (2020), this lemma can be easily generalized to continuous action space.

We are now ready to formally prove the lower bound.

Proof of Theorem 13. If $\mathrm{Reg}(\mathbb{A},\mathcal{M})\ge\frac{1}{48}L^{1/4}T^{3/4}$, the theorem holds trivially. Therefore, suppose $\mathrm{Reg}(\mathbb{A},\mathcal{M})<\frac{1}{48}L^{1/4}T^{3/4}$. Consider the case when the optimal price $p=\frac{1}{8}$. Partition $[\frac{3}{4},\frac{7}{8}]$ uniformly into subintervals of length $\lceil T^{-1/4}\rceil$. Formally, consider intervals $(x_j,x_{j+1}]$ where $x_j=$

[†††] Here the choice of $\frac{2}{3}$ and $\frac{1}{3}$ is not critical for our proof, and can be replaced with other constants.

$\frac{7}{8} - L^{-3/4}T^{-1/4}j$ for each $1 \le j \le m := \lfloor \frac{1}{8} L^{3/4}T^{1/4} \rfloor$. By Lemma 26, since $[x_j, x_{j-1}] \subset [\frac{3}{4}, \frac{7}{8}]$ for each $j \in [m]$, we have

$$\mathbb{E}_{R_p}\left[N\left(\mathbb{A}, \frac{3}{4}, \frac{7}{8}\right)\right] = \mathbb{E}_{R_p}\left[\sum_{j=1}^{m} N\left(\mathbb{A}; x_j, x_{j-1}\right)\right] = m \cdot \Omega\left(L^{-1/2}T^{1/2}\right) = \Omega\left(L^{1/4}T^{3/4}\right).$$

By our choice of $p = \frac{1}{8}$, the regret per round is $\Omega(1)$ when the explored price remains in $[\frac{3}{4}, \frac{7}{8}]$, thus

$$\mathrm{Reg}(\mathbb{A}, \mathcal{M}) \ge \mathrm{Reg}(\mathbb{A}, R_p) \ge \Omega\left(\mathbb{E}_{R_p}\left[N\left(\mathbb{A}; \frac{3}{4}, \frac{7}{8}\right)\right]\right) = \Omega(L^{1/4}T^{3/4}),$$

and the proof completes. □

III.4.3. Proof of Lemma 26 We split the proof of this lemma into two claims. First note that, in general, it is not true that all entrances in a low-regret decision tree are confident. In fact, having a small fraction of non-confident entrances may not affect the regret by too much. However, we can show that the *majority* of entrances should be confident.

Claim 4. *Let V_c be the set of confident entrances in $\mathbb{A}$ and v_{ent} be the random entrance node, then $\mathbb{P}_\chi(v_{\text{ent}} \in V_c) \ge \frac{1}{2}$ for any $\chi \in \{R, B\}$.*

Proof. Recall that v_{ent} is the (random) entrance node for a fixed interval $[a, b]$. We crucially observe that the two reward functions R, B are exactly identical for prices greater than b, so for any set of entrance nodes U, it holds $\mathbb{P}_R[v_{\text{ent}} \in U] = \mathbb{P}_B[v_{\text{ent}} \in U]$. For a contradiction, we assume $\mathbb{P}_R[v_{\text{ent}} \in \bar{V_c}] = \mathbb{P}_B[v_{\text{ent}} \in \bar{V_c}] \ge \frac{1}{2}$.

Let V be the set of all entrance nodes and

$$V_R := \left\{u \in V : \ \mathbb{P}_B[f_u(\ell) = R] > \frac{1}{3}\right\} \quad \text{and} \quad V_B := \left\{u \in V : \ \mathbb{P}_R[f_u(\ell) = B] > \frac{1}{3}\right\}.$$

By definition of V_c, if $u \in V \backslash V_c$ then either $\mathbb{P}_R[f_u(\ell) = B] \le \frac{1}{3}$ or $\mathbb{P}_B[f_u(\ell) = R] \le \frac{1}{3}$, so $\bar{V_c} = V_R \cup V_B$. Since $\mathbb{P}_B[v_{\text{ent}} \in \bar{V_c}] \ge \frac{1}{2}$, we have either $\mathbb{P}_R[v_{ent} \in V_R] > \frac{1}{4}$ or $\mathbb{P}_R[v_{ent} \in V_B] > \frac{1}{4}$. We derive contradictions for each case separately.

- Suppose $\mathbb{P}_R[v_{ent} \in V_B] > \frac{1}{4}$ and consider an entrance $v_{ent} \in V_B$. Recall that a leaf is blue if it is $\frac{T}{4}$ levels below v_{ent} and its price is still inside $[a, b]$, in other words, the policy has been selecting prices in $[a, b]$ for $\frac{T}{4}$ rounds. Thus,

  $$\mathbb{E}_R[N(a, b)] \ge \mathbb{P}_R[v_{ent} \in V_B] \cdot \mathbb{P}_R[f(\ell) = B] \cdot \frac{T}{4} \ge \frac{1}{4} \cdot \frac{1}{3} \cdot \frac{T}{4} = \frac{T}{48}.$$

  By Lemma 27, writing $\Delta(x) = r^* - R(x)$ for any price x,

  $$Reg(\mathbb{A}, R) \ge \mathbb{E}_R[N(a, b)] \cdot \min_{x \in [a, b]} \Delta(x) > \frac{T}{48} \cdot \frac{1}{2} = \frac{T}{96}.$$

  For $T > 2^{12}L$, it holds $\frac{1}{96}T > \frac{1}{48}L^{1/4}T^{3/4}$, a contradiction!

- Suppose $\mathbb{P}_B[v_{ent} \in V_R] > \frac{1}{4}$ and consider an entrance $v_{ent} \in V_R$. Recall that by Lemma 25, we w.l.o.g. assumed the depth of v_{ent} is at most $\frac{T}{2}$ (in the decision tree corresponding to policy $\mathbb{A}$), and that a leaf is colored red if its price drops below a within $\frac{T}{4}$ rounds after reaching v_{ent}, we deduce that the depth of any red leaf is at most $\frac{3T}{4}$. Thus, there are at least $\frac{T}{4}$ levels below each red leaf, or alternatively, $\frac{T}{4}$ rounds after the price drops below a. It follows that
$$\mathbb{E}_B[N(0,a)] \geq \mathbb{P}_B[v_{ent} \in V_R] \cdot \mathbb{P}_B[f(\ell) = R] \cdot \frac{T}{4} > \frac{1}{4} \cdot \frac{1}{3} \cdot \frac{T}{4} = \frac{T}{48}.$$
By Lemma 27, the regret can be lower bounded as
$$Reg(\mathbb{A}, B) \geq \mathbb{E}_B[N(0,a)] \cdot \min_{x \in [0,a]} \{\Delta(x)\} \geq \frac{T}{48} \cdot L^{1/4} T^{-1/4} = \frac{1}{48} L^{1/4} T^{3/4},$$
a contradiction! □

The second claim asserts that if v is a confident $[a,b]$ entrance, then the expected depth D of the (v,a,b)-classifier is large. This result enables us to lower bound the number of rounds that the policy selects prices in $[a,b]$, which can later be translated into a lower bound of the regret incurred in this interval.

Claim 5. *Let v be a confident $[a,b]$-entrance and $\mathbb{T}_v := (\Omega', x', f')$ be the (v,a,b)-classifier. Define*
$$D : L(\Omega') \to \mathbb{R}$$
$$\ell \mapsto d(\ell) - d(v),$$
where recall that $d(\cdot)$ denotes the depth of a node in $\mathbb{A}$ (instead of the (v,a,b)-classifier). Then,
$$\mathbb{E}_R[D] = \Omega(L^{-1/2} T^{1/2}) \quad \textit{and} \quad \mathbb{E}_B[D] = \Omega(L^{-1/2} T^{1/2}).$$

Proof. Since $x_u \in [a,b]$ for any node u of $\mathbb{T} = \mathbb{T}_v$,
$$|R(x_u) - B(x_u)| \leq 2L \cdot L^{-3/4} T^{-1/4} = 2L^{1/4} T^{-1/4}.$$
Thus, $\Delta(R,B) = \max_{u \in \mathbb{T}_v} \{KL(R(x_u), B(x_u))\} \leq 2(2L^{1/4}T^{-1/4})^2 = 8L^{1/2}T^{-1/2}$. With $\alpha = 2/3, \beta = 1/3$, Theorem 15 implies
$$\mathbb{E}_R[D | v_{\text{ent}} \in V_c] = \frac{\frac{1}{3}\log 2}{\Delta(R,B)} = \Omega(L^{-1/2} T^{1/2}). \quad \square$$

Lemma 26 then follows immediately from the above two claims. In fact,
$$\begin{aligned}\mathbb{E}_R[N(\mathbb{A}; a, b)] &\geq \mathbb{E}_R[N(\mathbb{A}; a,b) | v \in V_c] \cdot \mathbb{P}_R[v \in V_c] \\ &\geq \mathbb{E}_R[D | v \in V_c] \cdot \mathbb{P}_R[v \in V_c] \\ &\geq \Omega\left(L^{-1/2} T^{1/2}\right) \cdot \frac{1}{2} = \Omega(L^{-1/2} T^{1/2}),\end{aligned}$$
and Lemma 26 follows.

III.4.4. Proof of Lemma 25 Let $X(t)$ be the price policy $\mathbb{A}$ selects at t and recall that R is the true reward function. W.l.o.g. assume T is even number. Define policy $\mathbb{B}$ to be exactly identical to $\mathbb{A}$ in rounds $1,2,...,\frac{T}{2}$, and selects the price $X(\frac{T}{2})$ in all rounds after $\frac{T}{2}$. We will show that $\text{Reg}(\mathbb{B},R) \le 2\text{Reg}(\mathbb{A},R)$.

We first explain the intuition behind the proof. Viewing these two policies as two decision trees, we couple each path in $\mathbb{A}$ with the corresponding path in $\mathbb{B}$, and compare the *ex post* regret on these two paths. Suppose $X(\frac{T}{2}) = x \le p^*$. By unimodality, for any $p \le x$, we have $R(p) \le R(x)$, so further decreasing the price leads to lower reward rates. Thus the regret of $\mathbb{B}$ after $\frac{T}{2}$ in this case is lower than that of $\mathbb{A}$. Next consider $X(\frac{T}{2}) = x \ge p^*$. In this case, $\mathbb{A}$ has been selecting prices in $[x,1]$ before $\frac{T}{2}$ due to the markdown constraint. By unimodality, for any price $p \ge x$ we have $R(p) \le R(x)$, so the regret of $\mathbb{B}$ incurred after time $\frac{T}{2}$ (wherein the policy always selects x) is no greater than the regret of $\mathbb{A}$ incurred before $\frac{T}{2}$. Thus in both cases we may bound the regret of $\mathbb{B}$ by that of $\mathbb{A}$.

We now formalize the above idea. For any s,t with $0 \le s < t \le T$ and price p, define $N_s^t(p,\mathbb{A})$ (resp. $N_s^t(p,\mathbb{B})$) to be the number of rounds that policy $\mathbb{A}$ (resp. $\mathbb{B}$) selects p in rounds $[s,t]$. For any price p denote $\Delta(p) = r^* - R(p)$. Then, by (the continuous version) of Lemma 27, we may decomposition the regret as

$$\text{Reg}(\mathbb{B},R) = \int_0^1 \mathbb{E}[N_{T/2}^T(x)] \cdot \Delta(x)\ dx = \int_0^{p^*} \mathbb{E}[N_{T/2}^T(x)] \cdot \Delta(x)\ dx + \int_{p^*}^1 \mathbb{E}[N_{T/2}^T(x)] \cdot \Delta(x)\ dx. \quad (24)$$

Part I. Consider the first term in (24). For each fixed $x \le p^*$. Conditional on $X(\frac{T}{2}) = x$, we have $N_{T/2}^T(x,\mathbb{A}) = \frac{T}{2}$ and that $\mathbb{A}$ will only select prices in $[0,x]$ after time $\frac{T}{2}$, thus

$$\int_0^x \mathbb{E}[N_{T/2}^T(p,\mathbb{A}) | X(\frac{T}{2}) = x]\ dp = \frac{T}{2} = \mathbb{E}[N_{T/2}^T(x,\mathbb{B}) | X(\frac{T}{2}) = x].$$

By unimodality, $\Delta(x) \le \Delta(p)$ for each $p \le x$, so

$$\mathbb{E}[N_{T/2}^T(x,\mathbb{B}) | X(\frac{T}{2}) = x] \cdot \Delta(x) \le \int_0^x \mathbb{E}[N_{T/2}^T(p,\mathbb{A}) | X(\frac{T}{2}) = x] \cdot \Delta(p)\ dp. \quad (25)$$

Further, conditional on $X(\frac{T}{2}) = x$, policy $\mathbb{B}$ never selects any price $p \ge x$ after time $\frac{T}{2}$, so

$$\mathbb{E}[N_{T/2}^T(p) | X(\frac{T}{2}) = x] = 0,$$

and

$$\int_0^x \mathbb{E}[N_{T/2}^T(p,\mathbb{B}) | X(\frac{T}{2}) = x]\ dp = \int_0^{p^*} \mathbb{E}[N_{T/2}^T(p,\mathbb{A}) | X(\frac{T}{2}) = x]\ dp.$$

Combining with (25), we deduce that for any $x \le p^*$,

$$\mathbb{E}[N_{T/2}^T(x,\mathbb{B})|X(\frac{T}{2})=x]\cdot\Delta(x) \le \int_0^{p^*} \mathbb{E}[N_{T/2}^T(p,\mathbb{A})|X(\frac{T}{2})=x]\cdot\Delta(p)\ dp.$$

Let $f(x)$ be the density function of $X(\frac{T}{2})$. By integrating over x on $[0,p^*]$,

$$\int_0^{p^*} f(x)\cdot\mathbb{E}[N_{T/2}^T(x,\mathbb{B})|X(\frac{T}{2})=x]\cdot\Delta(x)\ dx \le \int_0^{p^*} f(x)\Big(\int_0^{p^*} \mathbb{E}[N_{T/2}^T(p,\mathbb{A})|X(\frac{T}{2})=x]\cdot\Delta(p)\ dp\Big)\ dx. \tag{26}$$

We simplify each side of the above as follows. Note that $N_{T/2}^T(x) = \mathbf{1}(X(\frac{T}{2})=x)\cdot\frac{T}{2}$, so

$$\text{LHS of (26)} = \int_0^{p^*} \mathbb{E}[N_{T/2}^T(x,\mathbb{B})]\cdot\Delta(x)\ dx.$$

On the other hand by exchanging the order of integration,

$$\begin{aligned}\text{RHS of (26)} &= \int_0^{p^*} \Delta(p)\Big(\int_0^{p^*} f(x)\cdot\mathbb{E}[N_{T/2}^T(p,\mathbb{A})|X(\frac{T}{2})=x]\ dx\Big)\ dp\\ &\le \int_0^{p^*} \Delta(p)\Big(\int_0^{1} f(x)\cdot\mathbb{E}[N_{T/2}^T(p,\mathbb{A})|X(\frac{T}{2})=x]\ dx\Big)\ dp\\ &= \int_0^{p^*} \Delta(p)\cdot\mathbb{E}[N_{T/2}^T(p)]\ dp \le \mathrm{Reg}(\mathbb{A},R),\end{aligned}$$

where the last inequality follows from Lemma 27. Substituting these two simplifications into (26), we have

$$\int_0^{p^*} \mathbb{E}[N_{T/2}^T(x,\mathbb{B})]\cdot\Delta(x)\ dx \le \mathrm{Reg}(\mathbb{A},R). \tag{27}$$

Part II. We next consider the second term in (24). Suppose $X(\frac{T}{2}) = x \ge p^*$, then $\mathbb{B}$ selects prices in $[x,1]$ before round $\frac{T}{2}$, and will select x in all rounds in after $\frac{T}{2}$. Thus,

$$\int_x^1 \mathbb{E}[N_0^{T/2}(p,\mathbb{A})|X(\frac{T}{2})=x]\ dp = \frac{T}{2} = \mathbb{E}[N_{T/2}^T(x,\mathbb{B})|X(\frac{T}{2})=x].$$

Since $x \ge p^*$, by unimodalty we have $\Delta(p)\ge\Delta(x)$ for any $p \ge x$. Moreover, due to the markdown constraint $\mathbb{A}$ never selected prices in $[0,x]$ before $\frac{T}{2}$, so

$$\int_{p^*}^1 \mathbb{E}[N_0^{T/2}(p,\mathbb{A})|X(\frac{T}{2})=x]\cdot\Delta(p)\ dp = \int_x^1 \mathbb{E}[N_0^{T/2}(p,\mathbb{A})|X(\frac{T}{2})=x]\cdot\Delta(p)\ dp \ge \mathbb{E}[N_{T/2}^T(x,\mathbb{B})|X(\frac{T}{2})=x]\cdot\Delta(x).$$

Integrating over prices x in $[p^*,1]$, we obtain

$$\int_{p^*}^1 f(x)\Big(\int_{p^*}^1 \mathbb{E}[N_0^{T/2}(p,\mathbb{A})|X(\frac{T}{2})=x]\cdot\Delta(p)\ dp\Big)\ dx \ge \int_{p^*}^1 f(x)\cdot\mathbb{E}[N_{T/2}^T(x,\mathbb{B})|X(\frac{T}{2})=x]\cdot\Delta(x)\ dx. \tag{28}$$

We now analyze each side of the above inequality respectively. By the same argument as in Part I, we may simplify the RHS as

$$\text{RHS of (28)} = \int_{p^*}^{1} \mathbb{E}[N_{T/2}^{T}(x, \mathbb{B})] \cdot \Delta(x)\ dx.$$

On the other hand,

$$\begin{aligned}\text{LHS of (28)} &= \int_{p^*}^{1} \Delta(p) \Big(\int_{p^*}^{1} \mathbb{E}[N_0^{T/2}(p, \mathbb{A}) | X(\frac{T}{2}) = x] \cdot f(x)\ dx \Big)\ dp \\ &\leq \int_{p^*}^{1} \Delta(p) \Big(\int_{0}^{1} \mathbb{E}[N_0^{T/2}(p, \mathbb{A}) | X(\frac{T}{2}) = x] \cdot f(x)\ dx \Big)\ dp \\ &= \int_{p^*}^{1} \Delta(p) \cdot \mathbb{E}[N_0^{T/2}(p, \mathbb{A})]\ dp \leq \text{Reg}(\mathbb{A}, R).\end{aligned}$$

Substituting into (28), we obtain

$$\int_{p^*}^{1} \mathbb{E}[N_{T/2}^{T}(x)] \cdot \Delta(x)\ dx \leq Reg(\mathbb{A}, R). \tag{29}$$

The proof of Lemma 25 then follows immediately by combining (27), (29) and substituting into (24). □

III.5. Dynamic Pricing with Markup Penalty

The markdown pricing problem can be viewed as dynamic pricing problem where each mark-up incurs an infinite penalty, and we have just shown a tight $\tilde{O}(T^{3/4})$ regret bound (Kleinberg (2005)). On the other hand, if prices can oscillate for free, the problem becomes ordinary Lipschitz bandits when $I = \infty$, which is known to admit an $\tilde{O}(T^{2/3})$ regret. Hence we arrive at a natural question:

If the penalty for each mark-up is finite, can we improve upon the $\tilde{O}(T^{3/4})$ bound for markdown pricing?

More precisely, given finite markup penalty, can we interpolate its regret bound between $\tilde{O}(T^{3/4})$, the bound in the presence of penalty, and $\tilde{O}(T^{2/3})$, the bound for zero markup cost?

We provide an affirmative answer to this question. We organize this section as follows. We first show that the regret bound can be improved to $\tilde{O}(T^{2/3})$, if $O(\log T)$ number of markups is allowed. Then we proceed to introduce the problem of dynamic pricing with markup penalty (DPMP), and derive immediately a tight regret bound for DPMP using the results established so far.

Algorithm 8 Geometric Successive Elimination Policy GSE$_{s,\varepsilon}$.

```
Input: $s, \varepsilon > 0$.
Initialize: $A \leftarrow \{i \cdot s | 0 \le i \le s^{-1}\}, \ell \leftarrow 2 \log \varepsilon^{-1}$.    ▷ Discretize price space into *arms*.
for $j = 0, 1, ..., \ell$ do    ▷ Exploration phase consists of $\ell$ *cycles*.
    for each $a \in A$ in decreasing order do
        Select $a$ for $2^j$ times in a row and observe rewards $X_{2^j}(a), ..., X_{2^{j+1}-1}(a)$.
        $\bar{\mu}(a) \leftarrow (2^{j+1} - 1)^{-1} \sum_{\tau=1}^{2^{j+1}-1} X_\tau(a)$.    ▷ Empirical mean.
        $[LCB(a), UCB(a)] \leftarrow [\bar{\mu}(a) - \sqrt{\frac{\log T}{2^{j+1}-1}}, \bar{\mu}(a) + \sqrt{\frac{\log T}{2^{j+1}-1}}]$.    ▷ Confidence interval.
    Eliminate $a$ from $A$ if there exists $a' \in A$ s.t. $UCB(a) < LCB(a')$.    ▷ Elimination.
Select any $a \in A$ henceforth.    ▷ Exploitation phase.
```

III.5.1. Dynamic Pricing with Few Markups Recall that for non-markdown pricing (i.e. Lipschitz bandits), a tight $\tilde{O}(T^{2/3})$ regret bound can be achieved by applying the Successive Elimination (SE) policy on a suitably discretized price space. However, such a policy may mark up for $\Omega(T^{2/3})$ times. As the cornerstone of this section, we first present a simple adaptation of the SE policy that achieves the same regret, $\tilde{O}(T^{2/3})$, but only marks up $O(\log T)$ times.

As in the SE policy, our *Geometric Successive Elimination* (GSE) policy (see Algorithm 8) discretizes the price space into *arms* and splits the time horizon into *cycles*. Different than the classical version where each cycle has the same length, in GSE the cycle lengths are given by a geometric sequence. In the j-th cycle, GSE sequentially selects each alive arm (i.e. not eliminated) for 2^j times consecutively, and at the end of each cycle eliminates the arms whose confidence intervals are dominated by some other arm. Specifically, the number of cycles that suffices to achieve the desired regret bound is only $O(\log T)$, in other words, GSE only marks up for $O(\log T)$ times. We formally state this result below.

Theorem 16 **(Pricing Policy with Few Markups)**. *Denote* GSE$_{s,\varepsilon}$ *the Geometric Successive Elimination Policy with parameters* $s, \varepsilon > 0$. *Then for* $s = L^{-2/3}T^{-1/3}$ *and* $\varepsilon = L^{1/3}T^{-1/3}$,

$$\mathrm{Reg}(\mathrm{GSE}_{s,\varepsilon}, \mathcal{F}_L) = O(L^{1/3}T^{2/3}\sqrt{\log T}).$$

We first describe the proof at a high level. The regret can be decomposed into the discretization error εT and the regret on the discretized instance. To bound the latter, we show that for any arm, the more suboptimal, the faster it gets eliminated (Lemma 28). Therefore, any alive arm at the end of the exploration phase is nearly optimal.

To formalized the above idea we need the following lemma. Throughout this section we denote $\Delta_i = r_{\max} - R(a_i)$ and N_i', N_i'' to be the number of rounds that arm a_i is selected in the exploration and exploitation phase respectively.

Lemma 28. *Suppose $R \in \mathcal{F}_L$ is the true reward function. For each $i \le s^{-1}$, Let $\mathcal{E}$ be the event that $N_i' \le \min\{T, 16\Delta_i^{-2}\log T\}$ for all $0 \le i \le s^{-1}$, then $\mathbb{P}[\mathcal{E}] \ge 1 - T^{-3}$.*

Proof. Suppose $i^* \in \arg\max_{0 \le k \le s^{-1}} R(k)$. For any $m \ge 1$, let $\bar{\mu}_m(i)$ be the mean reward of arm a_i when it is selected for m times. By Hoeffding inequality (Lemma 14), w.p. $1 - T^{-2}$,

$$LCB(a_{i^*}) = \bar{\mu}_m(i^*) - \sqrt{\frac{\log T}{m}} \ge R(i^*) - 2\sqrt{\frac{\log T}{m}},$$

and

$$UCB(a_i) = \bar{\mu}_m(i) + \sqrt{\frac{\log T}{m}} \le R(i) + 2\sqrt{\frac{\log T}{m}}.$$

If a_i is not eliminated, then $UCB(a_i) \ge LCB(a_{i^*})$, thus

$$R(a_{i^*}) - 2\sqrt{\frac{\log T}{m}} \le R(a) + 2\sqrt{\frac{\log T}{m}},$$

i.e. $m \le 16\Delta_i^{-2}\log T$, and the proof completes. □

Proof of Theorem 16. We first apply Lemma 27 to decompose the regret as follows. Let $a_i = i \cdot s$ be the i-th arm. Denote $r_{\max} = \max_{0 \le k \le s^{-1}} R(a_k)$ and $r^* = \max_{x \in [0,1]} R(x)$ the optimal reward rate in the discretized and continuous instance respectively. Then,

$$\begin{aligned}
\mathrm{Reg}(\mathrm{GSE}_{s,\varepsilon}, R) &= r^*T - r(\mathbb{A}, R) \\
&= \big(r^*T - r_{max}T\big) + \big(r_{max}T - r(\mathbb{A}, R)\big) \\
&\le LsT + \sum_{0 \le i \le s^{-1}} \mathbb{E}(N_i' + N_i'') \cdot \Delta_i && \text{By Lemma 27} \\
&= LsT + \sum_{i:\Delta_i \ge \varepsilon} \mathbb{E}N_i' \cdot \Delta_i + \sum_{i:\Delta_i < \varepsilon} \mathbb{E}N_i' \cdot \Delta_i + \sum_{0 \le i \le s^{-1}} \mathbb{E}N_i'' \cdot \Delta_i \\
&\le LsT + \sum_{i:\Delta_i \ge \varepsilon} \mathbb{E}N_i' \cdot \Delta_i + \varepsilon T + \sum_{0 \le i \le s^{-1}} \mathbb{E}N_i'' \cdot \Delta_i. && (30)
\end{aligned}$$

We need the following lemma to bound the second and last term. By Lemma 28, the second term in (30) can be bounded as

$$\begin{aligned}
\sum_{i:\Delta_i \ge \varepsilon} \mathbb{E}N_i' \cdot \Delta_i &= \sum_{i:\Delta_i \ge \varepsilon} \Delta_i \cdot \big(\mathbb{P}(\mathcal{E}) \cdot \mathbb{E}[N_i'|\mathcal{E}] + \mathbb{P}(\overline{\mathcal{E}}) \cdot \mathbb{E}[N_i'|\overline{\mathcal{E}}]\big) \\
&\le \sum_{i:\Delta_i \ge \varepsilon} \Delta_i \cdot \big(\mathbb{P}(\mathcal{E}) \cdot 16\Delta_i^{-2}\log T + \mathbb{P}(\overline{\mathcal{E}}) \cdot T\big) \\
&\le \sum_{i:\Delta_i \ge \varepsilon} \Delta_i \cdot \big(16\Delta_i^{-2}\log T + T^{-2}\big) \\
&\le 16s^{-1}\varepsilon^{-1}\log T + o(1). && \text{Since there are } s^{-1} \text{ arms} \quad (31)
\end{aligned}$$

We next bound the last term in (30). Observe that each arm alive at the end of the exploration phase has been selected for at least $2^\ell = 2^{2\log\varepsilon^{-1}} = \varepsilon^{-2}$ times. On the other hand, Lemma 28 says conditional on $\mathcal{E}$, each arm can be selected for at most $16\Delta_i^{-2}\log T$ times. Thus, conditional on $\mathcal{E}$, for each arm i selected for exploitation, it holds $\varepsilon^{-2} \le 16\Delta_i^{-2}\log T$, i.e. $\Delta_i \le 4\sqrt{\log T}\varepsilon$. Therefore,

$$\begin{aligned}\sum_{0\le i\le s^{-1}} \mathbb{E}N_i'' \cdot \Delta_i &= \sum_{0\le i\le s^{-1}} \left(\mathbb{E}[N_i''|\mathcal{E}]\cdot\mathbb{P}[\mathcal{E}] + \mathbb{E}[N_i''|\overline{\mathcal{E}}]\cdot\mathbb{P}[\overline{\mathcal{E}}]\right)\cdot\Delta_i\\ &\le \sum_{0\le i\le s^{-1}} \mathbb{E}[N_i''|\mathcal{E}]\cdot\Delta_i + T^{-3}\cdot\sum_{0\le i\le s^{-1}} T\\ &\le 4\sqrt{\log T}\varepsilon\cdot T + o(1). \end{aligned} \tag{32}$$

Substituting (31),(32) into (30), we have

$$(30) \le 16\varepsilon^{-1}s^{-1}\log T + 4\sqrt{\log T}\varepsilon\cdot T + (sL+\varepsilon)T + o(1).$$

The proof follows by selecting $\varepsilon = L^{1/3}T^{-1/3}$ and $s = L^{-2/3}T^{-1/3}\sqrt{\log T}$. □

We conclude this subsection with a couple of observations. First, in this result we no longer require the reward functions to be unimodal. Further, when T is unknown, we may apply the doubling trick (folklore, see e.g. Slivkins (2019)) to achieve the same bound. Finally, compared to the $O\big(T^{2/3}(L\log T)^{1/3}\big)$ bound for Lipschitz Bandits, this bound is only weaker by $O(\log^{1/6} T)$.

III.5.2. Dynamic Pricing with Markup Penalty In practice, an increase in price usually results in a decrease in demand (Homburg et al. (2005), Malc et al. (2016), Rotemberg (2002)). In this section, we model this effect by a introducing new concept, the *Markup Penalty Index* (MPI). As opposed to imposing a penalty on each markup, previous work has considered how promotions may boost the demand (Ramakrishnan (2012)). Building upon our lower bound techniques in the pure markdown setting, we show a tight regret bound in terms of the MPI for unimodal Lipschitz families.

Random Utility Model. We introduce a stylized model that captures the penalty incurred by mark-ups. Each buyer has two states: active and inactive. At each round, each *active* buyer's valuation v is i.i.d. drawn from some fixed unknown distribution $\mathcal{D}$, and she decides to buy if the utility $u(p,v) := v - p > 0$ where p is the current price. Thus, the demand rate (i.e. fraction of buyers who buy) at price p is $\mathbb{P}[u>0] = \mathbb{P}_{v\sim\mathcal{D}}[v>p]$. To model the negative effect of markups, we assume that each time the seller marks up, a constant γ fraction of active buyers become *inactive* and set their valuations to 0 *permanently.* Thus, after k mark-ups, the proportion of active buyers drops to $(1-\gamma)^k$ hence the demand rate becomes $(1-\gamma)^k\mathbb{P}[v>p]$. Since $(1-\gamma)^k = 1-k\gamma+o(\gamma)$, compared to having no markup, the seller "loses" $\big(1-(1-\gamma)^k\big)=k\gamma+o(\gamma)$ fraction of demand at price p. This amounts to charging a fixed penalty, additive cost penalty for each markup.

Extreme Cases. If $\gamma=O(T^{-1/3})$, then each markup incurs at most $\gamma T=O(T^{2/3})$ loss over the entire time horizon. In this case, by Theorem 16, an $\tilde{O}(T^{2/3})+\gamma T\log T=\tilde{O}(T^{2/3})$ regret can be achieved. If $\gamma=\Omega(1)$, then each markup incurs a $\Omega(T)$ loss, hence the markup penalty effectively becomes a hard constraint, and the problem reduces to the "pure" markdown problem discussed in the previous sections. Thus, the interesting case is when the order of γT is between $T^{2/3}$ and T.

Definition 17 (MPI). The *Markup Penalty Index (MPI)* is a number in $[0,1]$ defined as $c=1+\log_T\gamma$, i.e. the unique c satisfying $\gamma T=T^c$.

As in the previous sections, we compare the performance of a policy against the optimal FPP. Since the optimal FPP always selects $r^*=\max_{p\in[0,1]}R(p)$ where R is the underlying reward function, it pays no markup cost. Thus, the regret of a policy can be decomposed into the markup penalty and the cost of selecting suboptimal prices. Our goal is to find a policy that minimizes the regret, formalized below.

Definition 18 (Regret with Markup Penalties). Suppose the MPI is $c\in[0,1]$. For any policy $\mathbb{A}$, let $r(\mathbb{A},R)$ be its expected total reward and $\nu(\mathbb{A})$ be its total number of markups, i.e. $\nu=\sum_{t=1}^{T}\mathbf{1}[\mathbb{A}(t)<\mathbb{A}(t+1)]$ where $\mathbb{A}(t)$ is the (random) price $\mathbb{A}$ selects at t. Define the *regret* to be

$$\mathrm{Reg}_c(\mathbb{A},R)=r^*T-r(\mathbb{A},R)+\mathbb{E}_R[\nu(\mathbb{A})]\cdot T^c.$$

The regret on a family $\mathcal{F}$ of reward functions is simply $\mathrm{Reg}_c(\mathbb{A},\mathcal{F})=\max_{R\in\mathcal{F}}\mathrm{Reg}_c(\mathbb{A},R)$.

Note that the unimodality assumption is not needed here, so Theorem 16 immediately implies the following upper bound on $\mathcal{F}_L$.

Corollary 2 **(Upper Bound for Lipschitz Reward Functions)**. *Let* $\mathrm{GSE}_{s,\varepsilon}$ *be the Geometric Successive Elimination Policy with parameters $s,\varepsilon>0$. Then for $s=L^{-2/3}T^{-1/3}$ and $\varepsilon=L^{1/3}T^{-1/3}$, we have* $\mathrm{Reg}_c(\mathrm{GSE}_{s,\varepsilon},\mathcal{F}_L)=O(L^{1/3}T^{2/3}\sqrt{\log T}+T^c\log T)$.

However, when c is large, the above regret bound becomes poor. Observe that if, in addition, all reward functions are unimodal (i.e. replace $\mathcal{F}_L$ with $\hat{\mathcal{F}}_L$), the Uniform-Elimination policy (Algorithm 11) achieves regret $\tilde{O}(T^{3/4})$. Thus if c is known, by choosing the better policy between GSE and UE we achieve the following regret for $\hat{\mathcal{F}}_L$.

Theorem 17. *Let $\mathbb{A}$ be the policy that chooses* $\mathrm{GSE}_{s,\varepsilon}$ *with $s=L^{-2/3}T^{-1/3},\varepsilon=L^{1/3}T^{-1/3}$ when $c\le 3/4$ and chooses* $\mathrm{UE}_{s,\delta}$ *with $\delta=\sqrt{2}T^{-1/4}(L\log T)^{1/4},s=\delta/2L$ otherwise. Then,*

$$\mathrm{Reg}_c(\mathbb{A},\hat{\mathcal{F}}_L)=\tilde{O}(T^{med\{\frac{2}{3},c,\frac{3}{4}\}})=\begin{cases}O(T^{2/3}(L\log T)^{1/3}), & \textit{if } c\le 2/3,\\ O(T^c\log T), & \textit{if } 2/3<c\le 3/4,\\ O(T^{3/4}(L\log T)^{1/4}), & \textit{else.}\end{cases}$$

Surprisingly, this bound for $\hat{\mathcal{F}}_L$ turns out to be almost optimal as stated in the following theorem.

Theorem 18 **(Lower Bounds for General MPI)**. *For any policy $\mathbb{A}$ that knows the time horizon T and the true MPI c,*

$$\mathrm{Reg}_c(\mathbb{A},\hat{\mathcal{F}}_L) = \begin{cases} \Omega(T^{2/3}), & \textit{if } c < 2/3 \\ \Omega(T^{c}), & \textit{if } c \in [2/3, 3/4] \\ \Omega(T^{3/4}), & \textit{if } c > 3/4. \end{cases}$$

Our results show that the markup penalty index (MPI) plays a critical role in the achievable regret, and that three regimes emerge (Table 8). First, when the MPI is low ($c < 2/3$), the penalty for marking up is dominated by the cost incurred for searching for the optimal price. The regret (both upper and lower) is thus $\tilde{\Theta}(T^{2/3})$ no matter how small the MPI is. Second, when the MPI is moderate ($c \in [2/3, 3/4]$), the markup penalty now dominates the search cost. In this regime, careful restriction of the use of markups allows the regret to be limited to $\tilde{\Theta}(T^c)$. Third and finally, when the MPI is high ($c > 3/4$), the story remains the same, and unfortunately the regret is even linear if $c = 1$ (corresponding to the pure markdown setting). However, if in addition to the Lipschitz assumption, we assume that (a) the reward function is unimodal and that (b) the time horizon is known, then a separate policy which never marks up is able to achieve regret $\tilde{\Theta}(T^{3/4})$. These two assumptions together form a necessary and sufficient set of conditions for achieving sub-linear regret in the pure markdown setting.

All results until now have assumed that the time horizon T is known in advance. This assumption is, in a sense, necessary. In fact, we show that if T is unknown, no policy achieves $o(T^c)$ regret.

To this end, we first formalize what it means to "not know T". So far in this work, when we say a policy $\mathbb{A}$ "has regret $O(T^\alpha)$", we mean that there exists a **family** of decision trees $\{\mathbb{A}_T : T = 1, 2, ..\}$ and constants C, T_0, where $\mathbb{A}_T$ has depth T, and[‡‡‡] $\mathrm{Reg}_c(\mathbb{A}_T, \mathcal{F}, T) \le CT^\alpha$ for all $T \ge T_0$.

When T is unknown, however, we can no longer treat T as an input parameter. Instead, we view a policy $\mathbb{A}$ as an *infinite-level* decision tree. In this case, "$\mathbb{A}$ has $O(T^\alpha)$" regret means there exists constants C, T_0 s.t. $\mathrm{Reg}_c(\mathbb{A}, \mathcal{F}, T) \le CT^\alpha$ for all $T \ge T_0$. The next result says if T is unknown, then any policy can not improve the regret $O(T^c)$ by a polynomial factor (in T).

Proposition 4 **(Lower Bound for Unknown T)**. *Let c be the MPI. Then there is a family $\mathcal{F}$ of two reward functions such that for any infinite-level decision tree $\mathbb{A}$, and any constants $\varepsilon, T_0 > 0$, there exists $T > T_0$ with $\mathrm{Reg}_c(\mathbb{A}, \mathcal{F}, T) > T^{(1-\varepsilon)c}$.*

[‡‡‡] We use $\mathrm{Reg}_c(\mathbb{A}, \mathcal{F}, T)$ to denote the regret of a policy $\mathbb{A}$ in T rounds.

Proof. Consider a family $\mathcal{F} = \{R, B\}$ of reward functions where $R(x) = 1 - x$ and $B(x) = x$ for $x \in [0,1]$. Note that the optimal prices are $p_R^* = 0$ and $p_B^* = 1$. For the sake of contradiction, suppose there exists some $\varepsilon \in (0,1)$ and $T_0 > 0$ and an infinite-level decision tree $\mathbb{A}$ s.t. $\mathrm{Reg}_c(\mathbb{A}, \mathcal{F}, S) \leq T^{(1-\varepsilon)c}$ for any time horizon $S > T_0$.

We outline our proof at a high level. Consider the event $\mathcal{E}_T$ that $\mathbb{A}(T) < \frac{1}{2}$, where we recall that $\mathbb{A}(T)$ is the price at T. Denote $\delta(T) := \mathbb{P}_B[\mathcal{E}_T]$. We first show that $\{\delta(T)\}_{T \in \mathbf{Z}, T \geq T_0}$ is a nonzero, vanishing sequence. Then we show that however small $\delta(T)$ is, we may always construct a time horizon $S = S(T)$, where the one-time markup penalty S^c is high enough so that the *expected* markup penalty (which is at least the product of $\delta(T)$ and S^c) dominates the target regret, $T^{(1-\varepsilon)c}$, implying a contradiction.

We now formalize the above intuition. We first show that $\delta(T)$ has to be vanishing to achieve $O(T^{(1-\varepsilon)c})$ regret.

Claim 6. *For any $T \geq T_0$, it holds $\delta(T) > 0$. Moreover, $\delta(T) \to 0$ as $T \to \infty$.*

Assuming this claim, then by definition of limit, there exists a finite $\hat{T}$ such that $\delta(T) \leq 2^{-\frac{c\varepsilon}{2}}$, i.e. $\delta(T)^{-\frac{2}{c\varepsilon}} \geq 2$, for all $T > \hat{T}$. For any $T > \hat{T}$, consider time horizon $S(T) = \delta(T)^{-\frac{2}{c\varepsilon}} T$. Suppose B is the true reward function and condition on $\mathcal{E}_T$. If $\mathbb{A}$ does not mark up after T, then an $\Omega(1)$-regret is incurred in each of the $(S(T) - T)$ future rounds. If $\mathbb{A}$ does mark up, then a $\delta(T) \cdot (S(T) - T)^c$ penalty is incurred. Thus, denoting by q the probability that $\mathbb{A}$ ever marks up after T (conditional on $\mathcal{E}_T$), the regret is lower bounded by

$$\begin{aligned}
\mathrm{Reg}_c\big(\mathbb{A}, B, S\left(T\right)\big) &\geq \delta(T) \cdot \left((1-q) \cdot \frac{1}{2} \cdot (S(T) - T) + q \cdot (S(T) - T)^c \right) \\
&\geq \frac{1}{2} \delta(T) \cdot (S(T) - T)^c \\
&\geq \frac{1}{2} \delta(T) \cdot (\delta(T)^{-\frac{2}{c\varepsilon}} - 1)^c \cdot T^c \\
&> \frac{1}{2} \left(\delta(T)^{-\frac{2}{c\varepsilon}} T \right)^{(1-\frac{\varepsilon}{2})c} && \text{since } \delta(T)^{-\frac{2}{c\varepsilon}} \geq 2. \\
&\geq \Omega\left(S(T)^{(1-\frac{\varepsilon}{2})c} \right), && \text{as } T \to \infty,
\end{aligned}$$

a contradiction. □

Proof of Claim 6. We first consider the first part. Since $p_R^* = 0$ and $\mathbb{A}$ admits sublinear regret for any time horizon $T > T_0$, we deduce that there is at least one node on level T with price less than $\frac{1}{2}$. Since every path has nonzero probability under both reward functions, it follows that $\delta(T) > 0$ for any $T \geq \hat{T}$.

The other part is more involved, so we start with intuition. Suppose B is the true reward function and $\delta(T)$ does not vanish. Then for some longer time horizon, say $S := T^2$, w.p. $\delta(T)$ the policy overshoots the optimal price $p_B^* = 1$, incurring either high regret or markup penalty, yielding a contradiction.

We now make the ideas formal. For a contradiction, suppose there exists a constant $K > 0$ and a sequence $\{T_j\}_{j \in \mathbf{Z}^+}$ with $T_j \to \infty$, s.t. $\delta(T_j) > K$ for all j. Consider time horizon $S_j := T_j^2$. Suppose the true reward function is B and condition on $\mathcal{E}_{T_j}$. Our argument is similar to Proposition 4. If $\mathbb{A}$ does not mark up after T_j, then an $\Omega(1)$ regret is incurred in each of the future $S_j - T_j$ rounds. Otherwise, if $\mathbb{A}$ does mark up, a markup penalty S_j^c is incurred. Thus, denoting p the probability that $\mathbb{A}$ ever marks up after T (conditional on $\mathcal{E}_T$), we have

$$\begin{aligned} \mathrm{Reg}_c(\mathbb{A}, B, S_j) &\geq \delta(T_j) \cdot \left((1-p) \cdot \frac{1}{2} \cdot (S_j - T_j) + p \cdot (S_j - T_j)^c\right) \\ &> \frac{1}{2} K \cdot (S_j - T_j)^c = \Omega(S_j^c), \qquad \text{as } j \to \infty, \end{aligned}$$

a contradiction to the fact that $\mathbb{A}$ has $O(T^{(1-\varepsilon)c})$ regret. □

We conclude this section by summarizing our results from a different lens: our results show that the MPI plays a critical role in the achievable regret, and that three regimes emerge (Table 8).

	Low MPI	Med. MPI	High MPI
	$(c < 2/3)$	$(2/3 \leq c \leq 3/4)$	$(c > 3/4)$
Lipschitz	$\tilde{\Theta}(T^{2/3})$	$\tilde{\Theta}(T^c)$	$\tilde{\Theta}(T^c)$
Lipschitz, unimodal, known T	$\tilde{\Theta}(T^{2/3})$	$\tilde{\Theta}(T^c)$	$\tilde{\Theta}(T^{3/4})$

Table 8 Upper and lower Regret bounds in three different regimes based on c, the markup penalty index (MPI).

III.5.3. Unknown MPI Recall that our tight regret bound for DPMP selects between the UE policy (Algorithm 11) and the GSE policy (Algorithm 8), depending on how c, the MPI, compares with $\frac{3}{4}$. In particular, we assumed that the MPI is known to the policy. We show that this assumption is indeed necessary for achieving $o(T^{3/4})$ bound. In other words, if c is unknown, then DPMP is as hard to manage as the pure markdown problem.

Recall that $\mathcal{M}$ is the set of "mountain curves" defined in Section III.4. In the face of markup penalty, we represent an instance as a tuple $(\mathcal{M}, c)$ where $\mathcal{M}$ is known but c is unknown to the policy.

Proposition 5 **(Lower Bound for Unknown MPI)**. *Let $\mathcal{I}$ be the family of only two instances: $(\mathcal{M}, c = 0)$ and $(\mathcal{M}, c = 1)$. For any policy $\mathbb{A}$ oblivious of the MPI c, it holds* $\mathrm{Reg}_c(\mathbb{A}, \mathcal{I}) \geq \frac{1}{96} L^{1/4} T^{3/4}$.

Proof. Suppose $\mathbb{A}$ satisfies $\mathrm{Reg}_c(\mathbb{A},\mathcal{I}) \leq \frac{1}{96}L^{1/4}T^{3/4}$, in particular, $\mathrm{Reg}_c(\mathbb{A},\mathcal{M},c=1) = \frac{1}{96}L^{1/4}T^{3/4}$. For any $R\in\mathcal{M}$, let $q(R)$ be the probability that $\mathbb{A}$ ever marks up under R, i.e.

$$q(R) = \mathbb{P}_R[\sum_{t=2}^{T}\mathbf{1}\big(\mathbb{A}(t) > \mathbb{A}(t-1)\big) \geq 1],$$

where we recall that $\mathbb{A}(t)$ is the price $\mathbb{A}$ selects in round t. Observe that $q(R) \leq \frac{1}{96}L^{1/4}T^{-1/4}$ for any R. In fact, if $q(R) > \frac{1}{96}L^{1/4}T^{-1/4}$, when the MPI is $c=1$, the total markup penalty would be $\frac{1}{96}L^{1/4}T^{-1/4}\cdot T^1 = \frac{1}{96}L^{1/4}T^{3/4}$, hence $\mathrm{Reg}_{c=1}(\mathbb{A},R) \geq \frac{1}{96}L^{1/4}T^{3/4}$, a contradiction.

To apply the $\Omega(T^{3/4})$ lower bound for the pure markdown problem, we next transform $\mathbb{A}$ into a pure markdown policy $\mathbb{A}'$. Loosely, policy $\mathbb{A}'$ behaves exactly the same as $\mathbb{A}$ until the first time that $\mathbb{A}$ marks up, whereupon $\mathbb{A}'$ will stay at the same price forever since. We formalize this idea below. A node v is called a *markup node* if $x_v > x_{par(v)}$ where we recall that x_v is the price at node v and $par(v)$ is the parent node of v. For each markup node v, relabel its all descendants (including itself) with price $x_{par(v)}$. Since the reward along each path is at most T, and $q(R) \leq \frac{1}{96}L^{1/4}T^{-1/4}$, we have

$$|\mathrm{Reg}_c(\mathbb{A},R) - \mathrm{Reg}_c(\mathbb{A}',R)| \leq q(R)\cdot T \leq \frac{1}{96}L^{1/4}T^{3/4}.$$

Therefore,

$$\begin{aligned}\mathrm{Reg}(\mathbb{A},R) &\geq \mathrm{Reg}(\mathbb{A}',R) - \frac{1}{96}L^{1/4}T^{3/4}\\ &\geq \frac{1}{48}L^{1/4}T^{3/4} - \frac{1}{96}L^{1/4}T^{3/4} && \text{By Theorem 13}\\ &= \frac{1}{96}L^{1/4}T^{3/4}. && \square\end{aligned}$$

III.6. Experiments

In this section, we compare the empirical performance of the UE policy described in Section III.2.2 with several alternative policies in the case of infinite inventory. Our results demonstrate that the UE policy is reasonably fast in convergence and robust to model misspecification that affect other parametric policies adversely.

III.6.1. Robustness Under Model Misspecification We first compare the performance of our policy Uniform Elimination (UE) with two Explore-Then-Commit (ETC) type policies. A generic ETC policy assumes certain parametric form of the underlying demand function (which is possibly incorrect) and consists of an exploration phase and an exploitation phase. In the exploration phase, the policy randomly selects two *sample prices* p_1, p_2 near the maximum price, each for sufficiently many times. At the end of the exploration phase, the policy estimates the true parameters from the observations, and commits to the optimal price of the estimated demand function throughout the exploitation phase. We provide more details below.

ETC-Policies. An ETC policy is specified by two parameters: $h \in [0,1]$ and $k \in \mathbf{N}$. It first uniformly draws two sample prices p_1, p_2 from $[1-h, 1]$ and $[1-3h, 1-2h]$, and then selects each for k times. At the end of the exploration phase, the policy computes a demand function from the assumed demand family that best fits the empirical mean demands $\bar{d}_i$ at each p_i. Formally,

- ETC_{Lin} fits a linear demand function $\hat{D}(x) = \hat{a} - \hat{b}x$ given by
$$\hat{b} = -\frac{\bar{d}_1 - \bar{d}_2}{p_1 - p_2}, \quad \hat{a} = \hat{b} \cdot p_1 + \bar{d}_1.$$
- ETC_{Exp} fits an exponential demand function $\hat{D}(x) = \exp(\hat{a} - \hat{b}x)$ given by
$$\hat{b} = -\frac{\log \bar{d}_1 - \log \bar{d}_2}{p_1 - p_2}, \quad \hat{a} = \hat{b} \cdot p_1 + \log \bar{d}_1.$$

Finally, note that the optimal price is $\frac{a}{2b}$ for $D(x) = a - bx$, and $1/b$ for $D(x) = e^{a-bx}$, so in the exploitation phase the ETC policy selects the optimal price of $\hat{D}$, given by $\hat{p}_{\mathrm{lin}} = \mathrm{Clip}_{[0,1]}(\frac{\hat{a}}{2\hat{b}})$ and $\hat{p}_{\mathrm{exp}} = \mathrm{Clip}_{[0,1]}(1/\hat{b})$ for each case[§§§].

One of the merits of the UE policy is robustness: different than ETC policies, it does not assume a certain functional form on the underlying demand function. We compare the regret of the above three policies on linear demand functions $D(x) = \{a - bx\}$ and exponential demand functions $D(x) = \{e^{c-dx}\}$ over 1000 independently randomly generated demand functions. In each epoch, we randomly generate demand functions by drawing

$$a \sim U(0,1), b \sim U(0,a), c \sim U(0,3), d \sim U(0,10).$$

Note that b is capped at a since otherwise the value of the linear function may be negative. Furthermore, we scale each demand function so that the maximum reward rate is 1.

For UE, we set the parameters to be order-optimal, $s = \delta = \frac{1}{4}T^{1/4}$, as proved in Theorem 11. For simplicity, we ignore the the Lipschitz constant L in selecting the policy's parameters. For both ETC policies, we set $h = 0.1$ and for fairness of comparison, we set k to be the same as in UE, i.e. $k = \delta^{-2} \log T$. Our results in Fig 4 demonstrate the following properties of the UE policy:

1. Fast convergence rate of regret: the regret of UE vanishes at an appreciable speed for both linear and exponential demand functions.
2. Robustness to model misspecification: UE has vanishing regret on both demand functions. In comparison, the regret of each ETC policy converges to 0 at a speed much faster than UE, but does not converge under the incorrect model assumption.

[§§§] For any a, b, x, $\mathrm{Clip}_{[a,b]}(x)$ is defined to be the median of a, b and x.

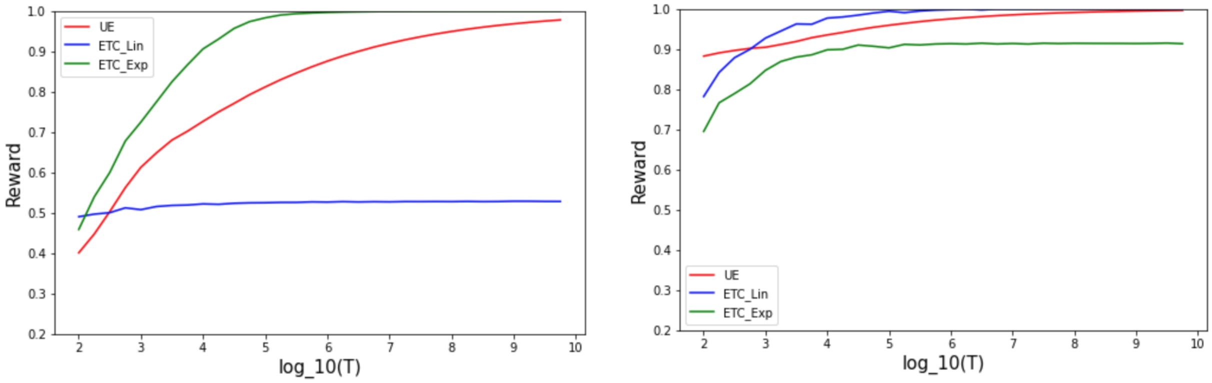

Figure 4 **Comparison between UE and ETC type policies on exponential and linear demand functions.**

III.6.2. Impact of the Lipschitz Constant Intuitively, as the Lipschitz constant L increases, reward function may change faster around the optimum, rendering the problem trickier. In our theoretical analysis, the $\tilde{O}(L^{1/4}T^{3/4})$ regret bound, further confirms this intuition. In this section we numerically investigate the influence of L.

Experiment Setup. We compare the performance of UE policy on several randomly generated families of exponential demand functions, each with a different range for Lipschitz constants.

We now describe how each curve in Figure 5 is obtained. Since we always scale the demand function so that the maximum reward rate is normalized to 1, the only parameter that matters in the demand function $D(x;a,b)=e^{a-bx}$ is b. For a fixed parameter range $[0,b_{\max}]$, we first compute the (minimal) Lipschitz constant $L=L(b_{\max})$ for the family $\mathcal{F}=\{D(x;0,b)|b\in[0,b_{\max}]\}$. Note that $L(b_{\max})$ increases as $b_{\max}$ increases. Then, we compute the average regret of our UE policy with the (order-) optimal hyper-parameters $s=L^{1/4}T^{3/4}$ and $\delta=L^{-3/4}T^{3/4}$ over 10^3 randomly drawn demand models $D(x;b)$ where $b\sim U(0,b_{\max})$. To generate a curve in Figure 5, we fix a T value and then compute the regret for integers $b_{\max}=1,2,...6$, and connect these 6 dots into a curve.

Finally, we let T vary from 10^5 to 10^9 and obtain 5 curves. We observed that for each T, the average regret is increasing in L. This matches our intuition since as $b_{\max}$ increases, the reward function becomes steeper around the peak, hence it is trickier to decide when to halt.

III.7. Conclusion

In this paper we showed tight regret bounds for markdown pricing under unknown demand model. Our regret bounds reveal that the markdown constraint adds significantly more complexity to dynamic pricing problems, since the corresponding regret bounds are asymptotically higher than without this constraint. Moreover, we introduced a new problem, dynamic pricing with markup penalty, that incorporates the negative effect of markups, and provided tight regret bounds. Finally,

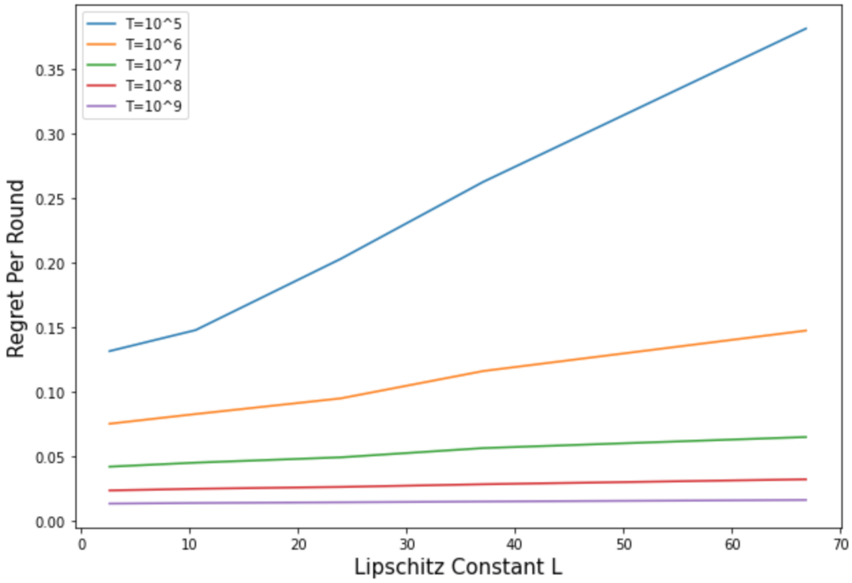

Figure 5 **How the regret changes for different Lipschitz constants.**

we showed through numerical experiments that our policy is robust to model misspecification and its regret vanishes rapidly.

Future Directions. This work opens up some directions in dynamic pricing for future research:

1. First, this work made minimal assumptions (Lipschitz and unimodal). In practice, however, demand functions usually take some simple functional forms. We showed how to improve these bounds when the reward function is twice-differentiable. An open problem in this direction is whether we can improve the regret bounds for specific families such as linear, exponential or logit demand functions. We provide some answers in the next chapter.
2. Second, our analysis compares against the best fixed price policy (FPP). For the infinite inventory version, the best FPP is optimal among all policies, but this is no longer true for the finite inventory version. An open question in this direction is, whether possible to analyze the regret against the best policy that knows the true demand function?
3. In this work we introduced a simple model that captures markup penalty and provided tight regret bounds. However, our stylized model fails to incorporate many practical aspects such as the heterogeneity in customer behaviors, and the impact of markup magnitude. We hope this work can open up a new direction in modelling the effect of markup.

Chapter IV Markdown Pricing Under Unknown Parametric Demand Models

In the previous chapter, we considered the markdown pricing problem where the underlying demand function is unknown, and showed a tight $T^{3/4}$ regret bound over T rounds under *minimal* assumptions of unimodality and Lipschitzness in the revenue function. This bound shows that the demand learning in markdown pricing is harder than regular pricing under unknown demand which suffers regret only of the order of $T^{2/3}$ under the same assumptions. However, in practice the demand functions are usually assumed to have certain functional forms, which may potentially render the demand- learning easier and lead to lower regret bounds. We investigate two fundamental questions in this chapter, assuming the underlying demand curve comes from a given parametric family:
(1) Can we improve the $T^{3/4}$ regret bound for markdown pricing under extra assumptions on the functional forms of the demand functions? We partially answered this question by showing a $T^{5/7}$ regret bound (Theorem 14) in the previous chapter, but is improvement possible with stronger assumptions?
(2) Is markdown pricing still *harder* than unconstrained pricing, under these additional assumptions? To answer these, we introduce a concept called *markdown dimension* that measures the complexity of the given family that contains the unknown true demand function. and present tight regret bounds under this framework, thereby completely settling the aforementioned questions.

IV.1. Introduction

Dynamic pricing under unknown demand has been extensively studied. Such problem arise naturally for new products, or for old products in new markets. Such problems are usually formulated as a Multi-Armed Bandit problem. While bandit problems have been well-understood *theoretically*, in practice however, we rarely see retailers deploy such policies. This is a largely because some practical constraints are often overlooked by those policies. For example, the prices may oscillate, which is highly undesirable. Quoting Bitran and Mondschein (1997) again,

> *"Customers will hardly be willing to buy a product whose price oscillates...Most retail stores do not increase the price of a seasonal or perishable product despite the fact that the product is being sold successfully".*

Dholakia (2021) has also pointed out in Harvard Business Review that

> *"Communicating a price increase to customers is never a pleasant task. It has the potential to stir customer service complaints, social media outrage, or simply lose customers altogether."*

More precisely, price increases may potentially create a manipulative image of the retailer and impact their ratings negatively. Luca and Reshef (2021) examined the relationship between price changes and the daily menu prices of some restaurants, and discovered that

"On average, a 1% price-increase leads to 3-5% decrease in online ratings."

Therefore, retailers may sometimes implicitly face a natural monotonicity constraint, that the prices can not go up. As defined in the the last chapter, a pricing policy that satisfies such a constraint is usually referred to as *markdown pricing* policy.

Markdown pricing is ubiquitous in retailing (Ramakrishnan (2012)). In practice, a retailer may use price markdowns to boost demands and hence increase revenues. For example, for fashion clothing, a retailer may start with a high retail price in the regular selling season, and then offer discounts in the clearance season, possibly over multiple rounds, to sell the remaining inventory. A successful markdown pricing strategy can have a considerable impact on the gross margins. A recent survey (Google (2021)) suggests that up to \$39 billion in value is being left on the table due to sub-optimal markdown pricing, and this number is just for one of many sectors of retail ("specialty" retail).

Thus motivated, in this work we consider the markdown pricing problem with unknown demand, under various assumptions. While *unconstrained* dynamic pricing under unknown demand has been extensively studied, little is known about *markdown* pricing under unknown demand. Recently, Jia et al. (2021) first considered this problem, under the most general case with only *minimal* assumptions for achieving meaningful performance guarantees. More precisely, they showed that unimodality and Lipschitzness in the *revenue* function (defined to be the price times mean demand) are necessary for attaining sublinear regret, and presented a tight $T^{3/4}$ regret bound under those assumptions. Noticeably, this bound is asymptotically higher than $T^{2/3}$, the known regret bound for *unconstrained* pricing, highlighting the extra complexity caused by the monotonicity constraint.

Nonetheless, in practice, demand functions are usually assumed to have certain parametric forms, such as linear, exponential or logit function. This motivates our first question:

Q1) Can we strengthen the $T^{3/4}$ regret bound for markdown pricing in this setting?

To see why such improvement is possible, we observe that the proof of the $T^{3/4}$ lower bound in the previous chapter considers pairs of "roof-shaped" revenue functions that are *completely* identical when the price is higher than some p, and diverging for prices lower than p. Thus, any reasonable policy has to carefully reduce the price, halting only when there is sufficient evidence for *overshooting* the optimal price.

However, this is not true in the parametric case. Take linear demand functions as an example. A policy may simply learn the slope and intercept of the underlying demand function at high prices, and then select the optimal price of the estimated demand function in all future rounds. This enables us to design a more powerful class of learn-then-earn type of policies. Now that we surmise that the $T^{3/4}$ regret can be improved under certain parametric assumption, we naturally arrive at our second question:

Q2) Is markdown pricing still harder than unconstrained pricing under these assumptions? Or more precisely, can we still show a *separation* between markdown and unconstrained pricing, under various parametric assumptions?

While one may answer these two questions for particular families such as linear family, the following is the real challenge.

Q3) Can we find a general framework to unify the regret bounds for different *categories* of families, rather than specific results for specific families?

In this work, we propose such a framework, by introducing a complexity index called *markdown dimension*, that captures the hardness of performing markdown pricing on a given family that contains the unknown true demand function. Under this framework, we provide efficient markdown policies for each dimension, which we also show to be *best* possible, thereby completely settling the problem of markdown pricing under unknown demand.

IV.1.1. Our Contributions.

In this work, we make the following contributions.

1. **New Complexity Measure of Demand Families:** We introduce a new concept called *markdown dimension* denoted by d, that captures the complexity of performing markdown pricing on a family, answering the third research question. Within this framework, we provide a complete settlement of the problem, as specified below.
2. **Markdown Policies with Theoretical Guarantees:** For each finite $d \geq 0$, we present a efficient markdown pricing policy. Our policies proceed in *phases*, wherein the seller learns the demand by selecting prices at suitable spacing to estimate the true parameter and then makes conservative decisions. We show that for $d=0$ and $d \geq 1$, our policies achieve regret $O(\log^2 T)$ and $\tilde{O}(T^{\frac{d}{d+1}})$ respectively, settling our first research question.
3. **Tight Minimax Lower Bound:** We complement our upper bounds with a matching lower bound for each markdown dimension. More precisely, we show that $\Omega(\log^2 T)$ regret is tight for dimension $d=0$, which *separates* it from the $O(\log T)$ regret bound without this monotonicity constraint. For finite $d \geq 1$, we show a $\Omega(T^{d/(d+1)})$ lower bound, which not only matches our upper bound (up to logarithmic factors) but is also asymptoticly higher than the tight $\tilde{\Theta}(T^{1/2})$

bound (see Broder and Rusmevichientong (2012)) without the markdown constraint, settling our second question.

4. **Impact of Smoothness:** We go further in refining our bounds and investigate the impact of smoothness of the revenue function around the optimal price, and extend our upper bounds for a generalization of smoothness that we call the sensitivity parameter $s \geq 2$. For both finite and infinite d, we obtained decreasing upper bounds as s increases from 2. Moreover for $d = \infty$, our tight $T^{\frac{2s+1}{3s+1}}$ regret bound is asymptoticly higher than that for unconstrained pricing, whose optimal regret is known to be $T^{\frac{s+1}{2s+1}}$ (Auer et al. (2007)).

The remainder of this paper is organized as follows: we conclude this section with a summary of the related literature. We then formally describe our model and assumptions in Section IV.2, and then state our policies and main results in Section IV.3.

IV.1.2. Previous Work The present work falls into two primary streams of work: dynamic pricing and multi-armed bandits. As mentioned above, the distinguishing feature of our work is the combination of a markdown constraint with a bandit-style (i.e. regret minimization) analysis. Other important dimensions along which to contrast this work with the extant literature include: whether the underlying demand function is assumed to come from a parametric family (this work is non-parametric), whether infinite inventory is assumed (this work allows for a particular regime of finite inventory), and whether it is assumed that a prior distribution for the demand functions is given (this work does not).

Dynamic Pricing: Gallego and Van Ryzin (1994) characterized the optimal pricing policy when the demand function is known. Kleinberg and Leighton (2003) studied a revenue maximization problem for a seller with an unlimited supply of identical goods, and obtained tight regret bounds under different models of buyers. Besbes and Zeevi (2009) studied the dynamic pricing problem under finite inventory in a finite selling period. Their benchmark regret function is the optimal pricing algorithm which is non-adaptive and whose expected sales is at most the inventory level. They presented an algorithm which achieves nearly optimal regret bounds. Subsequently, Wang et al. (2014) improved their results by showing a matching lower bound. Later, Babaioff et al. (2015) and Badanidiyuru et al. (2013) considered a more practical scenario where the inventory is finite. Other works that formulate dynamic pricing as MAB include Bastani et al. (2019), Hu et al. (2016), Chen and Farias (2018), Lei et al. (2014), Keskin and Zeevi (2014), den Boer and Zwart (2013), Liu and Cooper (2015), Farias and Van Roy (2010), Lobel (2020), Qiang and Bayati (2016), Papanastasiou and Savva (2017) and den Boer and Zwart (2015).

In practice, costs of implementing frequent price changes in a traditional retail setting can amount to a considerable portion of the seller's net margins. Thus motivated, Broder (2011) first formulated the demand learning problem with limited price changes and presented an $O(\sqrt{T})$ regret policy for parametric models using $O(\log T)$ price changes. Later, Perakis and Singhvi (2019) showed under stronger assumptions that the same regret may be achieved using $O(\log\log T)$ price-changes. Cheung et al. (2017) considered a given discrete demand functions and presented a regret bound that decreases in the number of allowed price-changes. Chen et al. (2020) considered the joint pricing and inventory management problem under limited price changes.

Orthogonal to the number of price changes, previous literature has also considered the *direction* of price changes. In practice, buyers usually have a *reference price* in mind, at which a higher (lower) price is considered a loss (gain), and customers are more sensitive to losses than to gains. Dynamic pricing with reference-price effects has been studied extensively in recent years, for example Nasiry and Popescu (2011), Heidhues and Kőszegi (2014), Wu et al. (2015), Hu et al. (2016) and Wang (2016). Recently, den Boer and Keskin (2020) considered the setting where the demand function is unknown.

As an important variant of the dynamic pricing problem, the *Markdown Pricing* problem has been extensively studied. The book chapter by Ramakrishnan (2012) and surveys by Elmaghraby and Keskinocak (2003) and den Boer and Zwart (2015) provide a through overview. Most previous work on markdown pricing assume a known demand function and focused on either empirical results (e.g. Smith and Achabal (1998), Heching et al. (2002)) or strategic customer behaviors (e.g. Yin et al. (2009), Boyacı and Özer (2010), Aviv and Vulcano (2012)). In our previous chapter, we introduced the markdown pricing under unknown demand function, and showed a $\tilde{\Theta}(T^{3/4})$ regret bound assuming the unknown revenue functions are Lipschitz and Unimodal.

Multi-armed Bandits (MAB): There exist several MAB variants that are similar to our problem, but without the markdown constraint. In the *Discrete Multi-armed Bandit* problem, the player is offered a finite set of arms, with each arm providing a random revenue from an unknown probability distribution specific to that arm. The objective of the player is to maximize the total revenue earned by pulling a sequence of arms (e.g. Lai and Robbins (1985)). Our pricing problem generalizes this framework by using an infinite action space $[0,1]$ with each price p corresponding to an action whose revenue is drawn from an unknown distribution with mean $R(p)$.

In the *Lipschitz Bandit* problem (e.g. Agrawal (1995)), it is assumed that each $x \in [0,1]$ corresponds to an arm with mean reward $\mu(x)$, and μ satisfies the Lipschitz condition, i.e. $|\mu(x) -$

$\mu(y)| \leq L|x-y|$ for some constant $L>0$. Kleinberg (2005) proved a tight $\tilde{\Theta}(T^{2/3})$ regret bound for one-dimensional Lipschitz Bandits. The lower bound was proved by considering a family of "bump curves": each curve is $\frac{1}{2}$ at all arms except in a small neighborhood of the "peak", where the mean reward is slightly higher elevated. Since these bump curves are unimodal, this lower bound carries over to the family we study.

Another closely-related variant of MAB is the *Unimodal Bandits* problem (Cope (2009), Yu and Mannor (2011), Combes and Proutiere (2014)). In addition to the Lipschitzness assumption, the reward function $\mu : [0,1] \to [0,1]$ is assumed to be unimodal. It is also assumed that there is a constant $L' > 0$ s.t. $|\mu(x) - \mu(y)| \geq L'|x-y|$ for all $x, y \in [0,1]$. Yu and Mannor (2011) proposed a binary-search type algorithm with regret $\tilde{O}(\sqrt{T})$.

Recently there is an emerging line of work on online learning with monotonicity constraint. Jia et al. (2021) considered the markdown pricing problem under unknown demand function and proved a tight $T^{3/4}$ regret bound under the minimal assumptions – Lipschitzness and unimodality on the revenue functions. Chen (2021) independently considered a special case where the inventory is infinite under the name *Monotone Bandits*, and obtained a subset of the results using different lower bound techniques. Gupta and Kamble (2019) and Salem et al. (2021) considered a more general online convex optimization problem where the actions sequence is required to be monotone.

IV.2. Model and Assumptions

We begin by formally stating our model. In this work we assume an unlimited supply of a single product. Given a discrete time horizon of T rounds, in each round t, the policy (representing the "seller") selects a price p_t (the particular interval $[0,1]$ is without loss of generality, by scaling). The demand D_t in this round is then independently drawn from a fixed distribution with *unknown* mean $D(p_t)$, and the policy receives revenue (or *reward*, which we will use interchangeably) p_t for each unit sold, and hence a total of $p_t \cdot D_t$ revenue in this round. The only constraint the policy must satisfy is the *markdown* constraint: $p_1 \geq \cdots \geq p_T$ almost surely.

The function $D(p)$ which maps each price p to the mean demand at that price is known as the *demand function*. For any policy π,[¶¶¶] demand function $D(\cdot)$, we use $r(\pi, D)$ to denote the expected total reward of π under D.

Rather than evaluating policies directly in terms of $r(\pi, D)$, it is more informative (and ubiquitous in the literature on multi-armed bandits) to measure performance using the notion of *regret*

[¶¶¶] For the sake of completeness, a *policy* is, formally, a time-indexed sequence of functions $\pi = \{\pi_t : ([0,1] \times [0,1])^{t-1} \to [0,1], t = 1, \ldots, T\}$, where each function π_t maps the prices selected and demands observed over the previous $t-1$ rounds to a price for round t.

with respect to a certain idealized benchmark. Specifically, since we assumed unlimited supply, when the true reward function is known, the seller simply always selects a revenue-maximizing price $p^*_D = \arg\max_{p\in[0,1]} p \cdot D(p)$ at each round, and we denote this maximal reward rate to be $r^*_D = \max_{p\in[0,1]} p \cdot D(p)$. The regret of a policy is then defined with respect to this quantity, and we seek to bound the *worst-case* value over a given family of demand functions.

Definition 19 (Regret). For any policy π and demand function D, define the *regret* of policy π under D to be $\mathrm{Reg}(\pi, D) := r^*_D \cdot T - r(\pi, D)$. For any given family $\mathcal{F}$ of demand functions, the *worst-case regret* (or simply *regret*) of policy π for family $\mathcal{F}$ is $\mathrm{Reg}(\pi, \mathcal{F}) := \sup_{D\in\mathcal{F}} \mathrm{Reg}(\pi, D)$.

IV.2.1. Basic Assumptions Now we state the common assumptions that all of our results rely on. A demand function $D(\cdot)$ is naturally associated with a *revenue function* $R(p) = p \cdot D(p)$, which we term more generally as the *reward function*. Sometimes it will be convenient to work directly with the revenue functions. In such cases, by abuse of notations, we may write $r(\pi, R)$ as the regret of under reward function is R and $r(\pi, \mathcal{F})$ the worst case regret under a family $\mathcal{F}$ of revenue functions.

Definition 20 (Optimal Price Mapping). Let $\mathcal{F}$ be a set of functions defined on some set $S \subseteq \mathbb{R}$. For any function $R : S \to \mathbb{R}$, let $M(R)$ be the subset of global maxima on S. The *optimal price mapping* of $\mathcal{F}$ is defined to be

$$\begin{aligned} p^* : \mathcal{F} &\to S \\ R &\to \inf M(R). \end{aligned}$$

By elementary topology, if the domain S is compact, then $M(R)$ is also compact, so the infimum of $M(R)$ can be attained, and hence $p^*(R)$ is also a global maximum of R. We first introduce a standard assumption (see e.g. Broder and Rusmevichientong (2012)), which assumes the derivative of R vanishes at $p^*(R)$.

Assumption 1 **(Vanishing Derivative)**. *We assume that every reward function R is differentiable on its domain, and moreover, $R'(p^*(R)) = 0$.*

In particular, this assumption holds if the reward function attains global its optimum in the interior of the domain. Under Assumption 1, the reward function changes gently around the optimal price, which intuitively leads to improved regret bounds. In fact, applying Taylor expansion on R around p^*, for sufficiently small ε it holds that

$$R(p^* + \varepsilon) = R(p^*) + R'(p^*) \cdot \varepsilon + \frac{1}{2} R''(p^*) \cdot \varepsilon^2 + o(\varepsilon^2),$$

i.e.

$$|R(p^*) - R(p^* + \varepsilon)| = \frac{1}{2} R''(p^*) \cdot \varepsilon^2 + o(\varepsilon^2).$$

Thus, if a policy overshoots the optimal price by ε, then only an $O(\varepsilon^2)$ loss is incurred in each round. We next introduce a *distributional* assumption.

Definition 21 (Subgaussian Random Variable). The *subgaussian norm* of a random variable X is

$$\|X\|_{\psi_2} := \inf\{c > 0 : \mathbb{E}[e^{X^2/c^2}] \le 2\},$$

and X is said to be *subgaussian* if $\|X\|_{\psi_2} < \infty$.

Assumption 2 **(Subgaussian noise)**. *There exists a constant $C_{sg} > 0$ such that under any true demand function and any price p, the random demand X at price p satisfies $\|X\|_{\psi_2} \le C_{sg}$.*

For example, this assumption is satisfied when the demand distributions are all Bernoulli, or when they are normal distributions with bounded variances.

Most of our upper bounds rely on the following standard concentration bound for subgaussian random variables (see e.g. Vershynin (2018)).

Theorem 19 **(Hoeffding's inequality)**. *Suppose $X_1, .., X_n$ are independent subgaussian random variables, then for any $\delta > 0$,*

$$\mathbb{P}\left[\sum_{i=1}^{n} (X_i - \mathbb{E}X_i) \ge \delta\right] \le \exp\left(-\frac{\delta^2}{2\sum_{i=1}^{n} \|X_i\|_{\psi_2}^2}\right).$$

IV.2.2. Measuring the Complexity of a Family Our goal in the rest of this section is to develop a novel concept, *markdown dimension*, that characterizes the complexity of a family of reward functions. In this subsection, we explain the high level ideas behind our definition.

Intuitively, the exploration-exploitation trade-off for markdown pricing becomes harder to manage as the given family becomes more complex. Consider, for example, linear demand functions. If each function takes the form $D(p; c) = 1 - cp$ where only $c \in (0, 1)$ is unknown and $p \in [0, 1]$, then the seller simply needs to estimate the (negative) slope c by sampling sufficiently many times at $p = 1$.

In contrast, if each function takes the form $D(p; a, b) = a - bp$ where *both* parameters a, b are unknown, then sampling at one price would *not* suffice. Rather, one needs to select (at least) two distinct prices to estimate a, b, thereby facing the following dilemma. Suppose the two prices $p < p'$ selected are far apart. Then, p may be far away from the unknown optimal price p^* since p^* may be close to p', resulting in a high regret. Otherwise, when those prices are close by, the demand learning requires a high volume of samples, which potentially also leads to a high regret.

Thus we reach a natural question: can we introduce a complexity index to measure the difficulty of performing markdown pricing on a given family, and then provide tight regret bounds in terms of this complexity index? A suitable choice of such complexity has been unexpectedly elusive. The first idea may be using the number of parameters to define the complexity. However, this is not well-defined, since there may be multiple ways to parametrize the same family, with possibly different numbers of parameters.

In this work, we propose such a complexity index, called *markdown dimension*, and provide nearly-optimal regret bounds in terms of the markdown dimension of the given family of demand functions. The formal definition relies on other two concepts, the *identifiability* of a family, and the *robustness* of a parametrization, which we introduce in the next two subsections.

IV.2.3. Identifiability Our notion of identifiability generalizes a key property of single-variable polynomials, that every degree-d polynomial can be uniquely determined by its values at *any* $(d+1)$ points. To present the formal definition, we first introduce a mapping which, for a fixed subset of prices, assigns each demand function a *profile* based on its values at those prices.

Definition 22 (Profile Mapping). Consider a set $\mathcal{F}$ of real-valued functions defined on $A \subseteq \mathbb{R}$. For any fixed $\mathbf{p} = (p_0, p_1, ..., p_d) \in A^{d+1}$, the *profile-mapping* with respect to $\mathbf{p}$ is defined as

$$\begin{aligned}\Phi_{\mathbf{p}} : \mathcal{F} &\to \mathbb{R}^d, \\ D &\mapsto (D(p_0), D(p_1), ..., D(p_d)).\end{aligned}$$

We may subsequently call $\Phi_{\mathbf{p}}(D)$ the *profile* of function D with respect to $\mathbf{p}$. Our notion of identifiability simply requires that every function in $\mathcal{F}$ be assigned a unique profile at any $(d+1)$ distinct points ("prices").

Definition 23 (Identifiability). The family $\mathcal{F}$ is *d-identifiable*, if for any $(d+1)$ *distinct* $p_0, p_1 ..., p_d \in S$, the profile mapping $\Phi_{p_0,...,p_d}$ is injective, i.e. distinct functions in $\mathcal{F}$ are mapped to distinct profiles.

In particular, if a family is d-identifiable, then for any distinct $p_0, p_1 ..., p_d$ the inverse profile-mapping $\Phi_{\mathbf{p}}^{-1} : \mathcal{R}_p \to \mathcal{F}$ exists, where $\mathcal{R}_p$ is the range of the mapping $\Phi_{\mathbf{p}}$.

IV.2.4. Robust Parametrization We first formally define a *parametrization*.

Definition 24 (Parametrization). An *order-m parametrization* for a family $\mathcal{F}$ of functions is any one-to-one mapping from a compact set $\Theta \subseteq \mathbb{R}^m$ to $\mathcal{F}$. Moreover, each value $\theta \in \Theta$ is called a *parameter*.

By abuse of notations, we may use $D(p;\theta)$ to denote the function $D(p)$ that parameter θ corresponds to. As a standard assumption (see e.g. Broder and Rusmevichientong (2012)), we also assume that the parameter set Θ to be compact, which leads to many favorable properties.

Assumption 3 **(Compact Domain)**. *The domain Θ of the parametrization is compact.*

Under this assumption, the demand functions in $\mathcal{F}$ are bounded, and thus we may without loss of generality also scale the range (i.e. target space) of those functions to be $[0,1]$.

Assumption 4 **(Smoothness)**. *The mapping $D:[0,1]\times\Theta\to\mathbb{R}$ is twice-differentiable and admits continuous second partial derivative. In particular, since $[0,1]\times\Theta$ is compact, under the above assumption, there exist constants $C^{(j)}>0$ such that the j-th derivative satisfies $|D^{(j)}(p,\theta)|\le C^{(j)}$ for any $(p,\theta)\in[0,1]\times\Theta$ and $j=0,1,2$.*

Recall that we previously defined the optimal price mapping p^* from $\mathcal{F}$ to the domain, $[0,1]$, of the reward functions. Now that we introduced a parametrization, by abuse of notation we may view the mapping p^* as being defined on $\Theta\subseteq\mathbb{R}^m$, which enables us to consider its Lipschitzness.

Assumption 5 **(Lipschitz Optimal Price Mapping)**. *The optimal price mapping $p^*:\Theta\to[0,1]$ is C^*-Lipschitz for some constant $C^*>0$.*

In particular, by compactness of Θ, a sufficient condition for the above to hold is that p^* admits continuous first-order partial derivatives. This assumption is hence satisfied by most "smooth" demand functions. For example, take linear demand function $D(p;c)=1-cp$ where $p,c\in[0,1]$. Then, the optimal price is $p^*(c)=\min\{\frac{1}{2c},1\}$, which can be easily seen to be 1-Lipschitz.

The final ingredient for robust parametrization is motivated by the following robustness of the *natural parametrization* $D(p;\theta)=\sum_{j=0}^{d}\theta_j p^j$ for polynomials. Consider any distinct prices $p_0,p_1,...,p_d$, and any $d+1$ real numbers $y_0,y_1,...,y_d$ representing, for example, the mean reward at each p_i. We may then uniquely determine a degree-d polynomial by solving the linear equation

$$\begin{bmatrix}1 & p_0 & p_0^2 \cdots & p_0^d\\ 1 & p_1 & p_1^2 \cdots & p_1^d\\ & & \vdots & \\ 1 & p_d & p_d^2 \cdots & p_d^d\end{bmatrix}\cdot\begin{bmatrix}\theta_0\\ \theta_1\\ \vdots\\ \theta_d\end{bmatrix}=\begin{bmatrix}y_0\\ y_1\\ \vdots\\ y_d\end{bmatrix}.$$

The matrix on the left-hand side is often referred to as the *Vandermonde* matrix, denoted $V_{\mathbf{p}}:=V(p_0,...,p_d)$. One can easily verify that when p_i's are distinct, $V_{\mathbf{p}}$ has non-zero determinant and is hence invertible, and hence

$$\theta=V_{\mathbf{p}}^{-1}\mathbf{y},$$

where $\mathbf{y}=(y_0,y_1,...,y_d)$. Next consider the effect of a perturbation on $\mathbf{y}$, in terms of the following *separability* parameter.

Definition 25. For any $\mathbf{p} = (p_0, ..., p_d) \in \mathbb{R}^{d+1}$, define $h(\mathbf{p}) := \min_{i \neq j} |p_i - p_j|$.
In words, this is the minimal distance between two distinct sample prices. To introduce the notion of robust parametrization, let us first consider a result that is specific to only polynomial demand functions.

Proposition 6. *There exist constants $C_1, C_2 > 0$ such that for any $\mathbf{p} \in \mathbb{R}^{d+1}$ with $0 < h(\mathbf{p}) \leq C_1$, any $\varepsilon \leq C_1$, and $\mathbf{y}, \hat{\mathbf{y}} \in \mathcal{R}_p$ with $\|\mathbf{y} - \hat{\mathbf{y}}\|_\infty \leq C_1$, it holds that*

$$\|V_{\mathbf{p}}^{-1}(\mathbf{y}) - V_{\mathbf{p}}^{-1}(\hat{\mathbf{y}})\|_\infty \leq C_2 \cdot \|\mathbf{y} - \hat{\mathbf{y}}\|_\infty \cdot h(\mathbf{p})^{-d}. \tag{33}$$

More concretely, let $D(p, \theta)$ be the underlying polynomial demand function, and $\mathbf{y} = V_p \cdot \theta$. Suppose $\hat{\mathbf{y}}$ corresponds to the empirical mean demands at prices $p_0, ..., p_d$. Then, one can estimate θ using the plug-in estimator $V_{\mathbf{p}}^{-1} \cdot \hat{\mathbf{y}}$. Proposition 6 then upper bounds the estimation error of this estimator, in terms of $h(\mathbf{p})$ and the difference between $\mathbf{y}$ and $\hat{\mathbf{y}}$. Our upper regret bounds are based on such estimation-error bounds.

As we will soon see, in order to achieve sublinear regret, the value $h = h(\mathbf{p})$ in our policy has to converges to 0, as T goes to infinity. Thus, the dependence on h crucially affects our regret bounds. Proposition 6 establishes a nice property for degree-d polynomials, that the estimation error increases in the order of $(\frac{1}{h})^d$, as $h \to 0^+$. We introduce the notion of robust parametrization by generalizing the above property beyond polynomials. Loosely, an order-d parametrization is *robust*, if it admits a similar error bound to (33).

Definition 26 (Robust Parametrization). An order-d parametrization $\theta : \Theta \to \mathcal{F}$ is *robust*, if

a) it satisfies Assumptions 3, 4 and 5, and
b) there exist constants $C_1, C_2 > 0$ such that for any $\mathbf{p} \in \mathbb{R}^{d+1}$ with $0 < h(\mathbf{p}) \leq C_1$ and any $y, y' \in \mathcal{R}_p$ with $\|\mathbf{y} - \mathbf{y}'\|_\infty \leq C_1$, it holds that

$$\|\Phi_{\mathbf{p}}^{-1}(\mathbf{y}) - \Phi_{\mathbf{p}}^{-1}(\mathbf{y}')\|_\infty \leq C_2 \cdot \|\mathbf{y} - \mathbf{y}'\|_\infty \cdot h(\mathbf{p})^{-d}.$$

IV.2.5. Markdown Dimension Now we are ready to define the markdown dimension.

Definition 27 (Markdown Dimension). The *markdown dimension* (or simply *dimension*) for a family $\mathcal{F}$ of functions, denoted $d(\mathcal{F})$, is the minimum integer $d \geq 0$ such that $\mathcal{F}$ is (i) d-identifiable, and (ii) admits a robust order-d parametrization. If no finite d satisfies the above conditions, then $d(\mathcal{F}) = \infty$.

For example, under mild assumptions, any family of degree-d polynomials is d-dimensional under the natural parametrization.

Proposition 7. *Let $\mathcal{F} = \{\sum_{j=0}^{d} \theta_j x^j \mid \theta \in \Theta\}$ where Θ is compact. Then, $\mathcal{F}$ is d-dimensional.*

We further illustrate our definition by considering the dimensions of some commonly used families. As the simplest family, one may verify that our definition of 0-dimensional family (under our assumptions) is equivalent to the *separable* family as defined in Section 4 of Broder and Rusmevichientong (2012). We provide more concrete examples below.

Proposition 8. *The following families are* 0*-dimensional:*

- *single-parameter linear demand functions: $\mathcal{F}_0 = \{1 - ax : b \in [a_{min}, a_{max}]\}$,*
- *exponential demand functions: $\mathcal{F}_1 = \{e^{1-ax} : b \in [a_{min}, a_{max}]\}$,*
- *logit demand functions: $\mathcal{F}_2 = \{\frac{e^{1-ax}}{1+e^{1-ax}} : b \in [a_{min}, a_{max}]\}$,*

Finally, we observe that by our definition, if a family of functions is not d-identifiable for any d, then it is infinite dimensional, as illustrated by the following example.

Proposition 9. *Let $\mathcal{F}$ be the set of all 1-Lipschitz functions on $[0,1]$, then $d(\mathcal{F}) = \infty$.*

In this work, for each finite $d = 0, 1, 2, ...$, we will propose an efficient markdown pricing policy for dimension d families, which we also prove to be the best possible theoretically.

IV.2.6. Sensitivity Consider the Taylor expansion of a reward function $R(x)$ around an optimal price p^*:

$$R(p) = R(p^*) + 0 + \frac{1}{2!}R''(p^*)(p - p^*)^2 + \frac{1}{3!}R^{(3)}(p^*)(p - p^*)^3 + ...$$

Suppose the first nonzero derivative is $R^{(k)}(p^*)$. Then, the higher k, the less the revenue is *sensitive* to overshooting (i.e. $p < p^*$). This motivates us to introduce the following concept, *sensitivity*, that measures how fast the revenue function changes around the optimal price.

Definition 28 (Sensitivity). A reward function is called *s-sensitive* if every function $R \in \mathcal{F}$ is $(s+1)$-differentiable, with $R^{(1)}(p^*(R)) = ... = R^{(s-1)}(p^*(R)) = 0$ and $R^{(s)}(p^*(R)) < 0$. A family $\mathcal{F}$ of reward functions is called *s-sensitive* if

a) every $R \in \mathcal{F}$ is s-sensitive,

b) it admits a parametrization $R(x;\theta)$ satisfying Assumptions 3, 4 and 5, and

c) there is a constant $C_6 > 0$ such that $R^{(s)}(p^*(R)) \le -C_6 < 0$ for any $R \in \mathcal{F}$.

One can easily verify the following.

Proposition 10. *Let $R(x;\theta) = \theta - |\frac{1}{2} - x|^s$ for $x \in [0,1]$, then $\{R(x;\theta) : \theta \in [\frac{1}{2}, 1]\}$ is an s-sensitive family.*

To utilize the sensitivity of a family, we will use a folklore result in the upper bound analysis.

Theorem 20 **(Taylor's Theorem with Lagrange Remainder)**. *Let $f : \mathbb{R} \to \mathbb{R}$ be $(m+1)$ times differentiable on an open interval (a,b). Then for any $x, x' \in (a,b)$, there exists some ξ with $(x-\xi)\cdot(x'-\xi) \le 0$ such that*

$$f(x') = f(x) + \frac{1}{1!}f'(x)(x'-x) + \ldots + \frac{1}{m!}f^{(m)}(x)(x'-x)^s + \frac{1}{(m+1)!}f^{(m+1)}(\xi)(x'-x)^{s+1}.$$

Theorem 20 implies a key property for any s-sensitive reward functions. Suppose R is s-sensitive, then for any $\varepsilon > 0$, we have

$$R(p^* + \varepsilon) = R(p^*) + \frac{R^{(s)}(\xi)}{s!}\varepsilon^s$$

where $\xi \in [p^*, p^* + \varepsilon]$. Since Θ is compact, there exists some constant $C_s > 0$ such that $|R^{(s)}(x,\theta)| \le C_s$ for any $x \in [0,1]$ and $\theta \in \Theta$. Thus,

$$|R(p^* + \varepsilon) - R(p^*)| \le \frac{C_s}{s!}|\varepsilon|^s.$$

Consequently, if a policy overshoots or undershoots the optimal price by ε, the regret *per round* is only $O(\varepsilon^s)$, which is asymptoticly lower than the per-round regret $O(\varepsilon^2)$ without the sensitivity assumption.

For unconstrained pricing (or continuum bandits), Kleinberg (2005) showed that a tight $T^{\frac{s+1}{2s+1}}$ regret for s-sensitive reward functions. In particular, for $s = 1$, the regret becomes $T^{2/3}$, which is strictly lower than the $T^{3/4}$ result. In this work we address a natural question: how does the regret bounds for markdown pricing change as s increases?

IV.3. Policies and Results

We give an overview of our policies and prove regret upper bounds for zero, finite (non-zero) and infinite dimensional families of demand functions. Moreover, we show that our policies achieves nearly-optimal regret by providing lower bounds for each of these regimes.

IV.3.1. Zero-Dimensional Family We start with the simplest case, 0-dimensional demand functions. We propose a policy called *Cautious Myopic* which proceeds by *phases* and makes *conservative* decisions. As opposed to the *optimism* in the face of uncertainty in UCB type policies, our policy adopts *conservatism* in the face of uncertainty.

More precisely, we partition the time horizon so that the j-th phase consists of $t_j := \lceil 2^j \log T \rceil$ rounds (for simplicity we assume T is a multiple of $\lceil \log T \rceil$), and thus in total there are $K = O(\log T - \log\log T)$ phases. In each phase, the policy estimates the true parameter θ^* using the observations from the last phase, and builds a confidence interval around θ^*, which depends on the number of length of this phase and also the constant C_{sg} as defined in Assumption 2. Then, in the next phase, the policy selects the *largest* optimal price of any parameter θ in the confidence interval. We write $t^{(j)} := \sum_{k=0}^{j} t_k$ and for convenience $t^{(0)} = 0$, and formally state this policy in Algorithm 9.

Algorithm 9 Cautious Myopic Policy.

```
Input: a family $\mathcal{F}$ of demand functions and time horizon $T$.
$p_1 \leftarrow 1$                                              ▷ Initialization
for $j = 1, ..., K$ do
    for $t = t^{(j-1)} + 1, ..., t^{(j-1)} + t_j$ do                ▷ Phase $j$ starts
        $x_t \leftarrow p_j$                                    ▷ Select $p_j$ for $t_j$ times in a row
        Observe realized demand $D_t$
    $\bar{d}_j = \frac{1}{t_j} \sum_{\tau=1}^{t_j} D_{t^{(j-1)}+\tau}$          ▷ Empirical mean demand in phase $j$
    $\hat{\theta}_j \leftarrow \Phi_{p_j}^{-1}(\bar{d}_j)$                  ▷ Estimate parameter
    $w_j \leftarrow 2C_{sg}\sqrt{\frac{\log T}{t_j}}$                    ▷ Width of the confidence interval
    $p_{j+1} \leftarrow \max\{p^*(\theta) : |\theta - \hat{\theta}_j| \le w_j\}$    ▷ Conservative estimation of the optimal price
```

By bounding the expected regret in each round using concentration bounds, we obtain our first following upper bound.

THEOREM 21 **(Zero-dimensional Upper Bound)**. *Let $\mathcal{F}$ be any 0-dimensional, s-sensitive family of demand functions. Then the Cautious Myopic (CM) Policy has regret*

$$\mathrm{Reg}(\mathrm{CM}, \mathcal{F}) = \begin{cases} O(\log^2 T), & \textit{if } s = 2, \\ O(\log T), & \textit{if } s > 2. \end{cases}$$

As we discussed earlier, many simple families of demand functions such as single-parameter linear or exponential demand functions satisfy $s = 2$, thereby having regret $O(\log^2 T)$.

It is worth noting that this bound is asymptotically higher than the $O(\log T)$ upper bound in the absence of the markdown constraint (Broder and Rusmevichientong (2012)). Intuitively, this is because the CM policy strikes a balance between the risk of overshooting (the optimal price) and getting close to the optimal price, by purposely distancing from the estimated optimal price. Is this trade-off optimal? In other words, can we achieve $o(\log^2 T)$ regret by taking more risk or being more conservative?

We answer this question by showing that CM is indeed optimal, that is, there is an $\Omega(\log^2 T)$ lower bound. Further, this result provides the first *separation* between the $O(\log T)$ regret for unconstrainted pricing and markdown pricing for 0-dimensional demand families.

Theorem 22 **(Zero-Dimensional Lower Bound)**. *For any $\theta \in \mathbb{R}$, define $D_\theta(x) = 1 - \theta x$ for $x \in [\frac{1}{2}, 1]$ and consider $\mathcal{F} = \{D(x;\theta) : \theta \in [\frac{1}{2}, 1]\}$. Then, $\mathcal{F}$ is* 0*-dimensional and for any policy π,* $\mathrm{Reg}(\pi, \mathcal{F}) = \Omega(\log^2 T)$.

IV.3.2. Finite-Dimensional Family Now we consider finite, nonzero dimensional families. Different from the zero-dimensional case, now the learner is no longer able to estimate the true parameter θ at a single price. Rather, for dimension d, the learner needs to collect demand samples at $d+1$ distinct *sample prices.* This, however, introduces extra regret, since the optimal price may lie *between* these sample prices.

Intuitively, a reasonable policy needs to trade off between the overshooting risk and the learning rate. If the gap is large, the policy may learn the parameter efficiently, but there is potentially a higher regret due to overshooting, in case the true optimal price lie between the sample prices. On the other side, if the gap is small then there is less risk of overshooting but a slower rate of learning.

We introduce our Iterative Cautious Myopic (ICM) Policy (Algorithm 10) that strikes such balance nearly optimally, as we will soon see from Theorem 23 and Theorem 24. The policy consists of m phases. In phase $j \in [m]$, the policy selects d *sample prices*, evenly spaced with distance h, and each for T_j times. Then, the policy estimates the optimal price based on the observed demands, and constructs a confidence interval $[L_j, U_j]$ centered at $\hat{p}_j$.

To determine the initial price p_{j+1} in the next phase, the policy considers the following three cases. Note that the last price that the policy selects in phase j is $p_j - dh$. We say a *good* event occurs, if $p_j - dh > U_j$. in which case we simply select the next sample price, p_{j+1}, to be U_j. In the *dangerous* event, the current price $p_j - dh$ is within the confidence interval, and we may have already overshot the optimal price. In this case, we can no longer select p_{j+1} to be U_j due to the markdown constraint. Instead, we select $p_{j+1} = p_j - dh$. Finally in the *overshooting* event, as the name suggests, our current price is already lower than the left endpoint L_j of the confidence interval, and hence with high probability we have overshot the optimal price. In this case, we immediately exit the exploration phase (i.e. the outer for-loop) and enter the exploitation phase, wherein the current price is selected in all future rounds.

Algorithm 10 Iterative Cautious Myopic Policy.

```
Input: $\mathcal{F}, m, \{T_j\}_{j\in[m]}, T$
$p_1 \leftarrow 1, L_0 \leftarrow 0, U_0 \leftarrow 1$                          ▷ Initialization
for $j = 1, 2, ...m$ do                                                         ▷ Phases
    for $k = 0, 1, ..., d$ do                     ▷ Sample at $(d+1)$ equi-distant prices
        Select price $p_j - kh$ for $T_j$ times.
        $\bar{D}_k \leftarrow \frac{1}{T_j}\sum_{i=1}^{T_j} D_i$                 ▷ Mean demand at $p_j - kh$
    $\hat{\theta} \leftarrow \Phi^{-1}_{p_j,...,p_j-dh}(\bar{D}_0, ..., \bar{D}_d)$            ▷ Estimate Parameter
    $w_j \leftarrow 2h^{-d} \cdot C_2 \cdot C_{sg}\sqrt{\frac{d\log T}{T_j}}$                  ▷ Width of confidence interval
    $L_j \leftarrow \min\{p^*(\theta) : \|\theta - p^*(\hat{\theta})\|_2 \le w_j\}$          ▷ Lower confidence bound
    $U_j \leftarrow \max\{p^*(\theta) : \|\theta - p^*(\hat{\theta})\|_2 \le w_j\}$          ▷ Upper confidence bound
    if $U_j \le p_j - dh$ then $p_{j+1} \leftarrow U_j$                             ▷ Good event
    if $U_j > p_j - dh \ge L_j$ then $p_{j+1} \leftarrow p_j - dh$                    ▷ Dangerous event
    if $p_j - dh < L_j$ then Break                                          ▷ Overshooting event
Select the current price in every future round                                 ▷ Exploitation
```

Theorem 23 **(Upper Bound for Finite $d \ge 1$)**. *For any $m = \tilde{O}(1)$, there exists suitable choice of $h > 0$ and $T_1 < ... < T_m$, such that the Iterative Cautious Myopic Policy Policy* ICM = ICM$(T_1, ..., T_m, h)$ *achieves regret* $\mathrm{Reg}(\mathrm{ICM}, \mathcal{F}) = \tilde{O}\left(T^{\rho(m,s,d)}\right)$ *where*

$$\rho(m, s, d) = \frac{1 + \left(1 + \frac{s}{2} + ... + (\frac{s}{2})^{m-1}\right) d}{\left(\frac{s}{2}\right)^m + \left(1 + \frac{s}{2} + ... + (\frac{s}{2})^{m-1}\right) \cdot (d+1)}.$$

In particular, for $s = 2$ and $m = \log T$, we have

$$\mathrm{Reg}\left(\mathrm{ICM}, \mathcal{F}\right) = \tilde{O}\left(T^{\frac{d}{d+1}}\right).$$

In contrast to the upper bound for zero-dimensional family where the regret is only logarithmic in T, for $d \ge 1$ the regret scales polynomially in T. We complement the upper bound with an nearly tight lower bound, up to $\log T$ factors, stated below.

Theorem 24 **(Lower Bound for Finite $d \ge 1$)**. *For any $d \ge 2$, there exists a d-dimensional family $\mathcal{F}$ of demand functions on* $[0, 1]$ *such that for any markdown policy π,*

$$\mathrm{Reg}(\pi, \mathcal{F}) = \Omega(T^{\frac{d}{d+1}}).$$

In our proof, for each $d \geq 1$ we consider a sub-family of $(d+1)$-degree decreasing polynomial demand functions – which is also d-dimensional – and show that there is a pair of such demand functions on which any policy suffers regret $\Omega(T^{\frac{d}{d+1}})$.

Markdown Dimension	Markdown	Unconstr. Pricing
$d = 0$	$\Theta(\log^2 T)$	$\Theta(\log T)$
$1 \leq d < \infty$	$\tilde{\Theta}(T^{d/(d+1)})$	$\tilde{\Theta}(\sqrt{T})$

Table 9 Regret bounds for markdown and unconstrainted pricing under unknown demand for $s = 2$.

We summarize our results for $s = 2$ in Table 9. We highlight our results in red, and emphasize that each entry corresponds to two results, an upper bound and a matching lower bound. Notation $\tilde{\Theta}$ means ignoring $\log T$ terms.

IV.3.3. Infinite Dimensional Family

For the infinite dimensional functions, it is more convenient to work with the mean revenue (or reward, which we use interchangeably) function $R(x) := x \cdot D(x)$ instead of the demand function. In particular, this reward function can be determined completely by $D(x)$, and thus the reward function $R(x)$ is unknown if and only if the demand function $D(x)$ is unknown. Further, it is straightforward to verify the following.

Fact. For any $0 \leq d \leq \infty$, a family of demand functions has dimension d if and only if its corresponding family of reward functions has dimension d.

Many previous work on dynamic pricing and multi-armed bandits focused on *infinite* dimensional families of demand functions. For example, it has been shown that for the family of Lipschitz demand functions an $\tilde{O}(T^{2/3})$ regret can be achieved (Kleinberg (2005), Broder and Rusmevichientong (2012)). Another well-studied setting is when the reward functions correspond to demand functions that are unimodal (Yu and Mannor (2011), Combes and Proutiere (2014)), where a binary search type policy achieves $\tilde{O}(T^{1/2})$ regret under an additional lower Lipschitz assumption.

In contrast to the *unconstrained* version, there is no *markdown* policy that achieves $o(T)$ regret on the family of Lipschitz reward functions (see Jia et al. (2021)). In fact, consider reward functions with possibly multiple local optima. Suppose a policy detects a local optimum at some high price p_{high}, then it faces a dilemma: if it stops at p_{high}, then a high regret is incurred since it may potentially earn rewards at a faster rate at some lower price. On the other side, if it does further reduce the price, it may be the case that no lower prices have as high reward as at p_{high}, and due to the markdown constraint, the policy may not increase the price back to p_{high}, leading to a high future regret.

It is worth noting that for finite dimensional families such dilemma is lifted, since by definition of dimension, the learner may infer whether or not a lower price has higher reward rates by simply collecting more samples at p_{high}. This is, however, not true for the Lipschitz family, since two Lipschitz reward functions that behave drastically differently at low prices may be completely identical at higher prices.

Nonetheless, Jia et al. (2021) showed that if the underlying demand functions are assumed to be Lipschitz and unimodal (which are both satisfied by many commonly used families), then a tight $\tilde{\Theta}(T^{3/4})$ regret is achievable. With this unimodal assumption, the markdown pricing problem essentially becomes finding the unique local optimum of the true revenue function. Specifically, their lower bound is derived on a family of Lipschitz reward functions where the reward rate may change abruptly at the peak.

Can the regret bound be improved if the reward functions are assumed to change smoothly? We answer this question by generalizing their result to incorporate the sensitivity parameter. Let $\mathcal{F}_s^U$ be the family of unimodal, s-sensitive reward functions.

Algorithm 11 Uniform Elimination Policy (UE$_{m,\Delta}$).

```
Input: $T, \Delta, m > 0$
Initialize: $L_{\max} \leftarrow 0$, $w \leftarrow 2C_{sg}\sqrt{\frac{\log T}{m}}$
for $j = 0, 1, 2, ..., \lceil \Delta^{-1} \rceil$ do                     ▷ Exploration phase starts
    $x_j \leftarrow 1 - j\Delta$
    Select price $x_j$ for the next $m$ rounds and observe rewards $Z_1^j, ..., Z_m^j$
    $\bar{\mu}_j \leftarrow \frac{1}{m}\sum_{i=1}^m Z_i^j$                          ▷ Compute mean rewards
    $[L_j, U_j] \leftarrow [\bar{\mu}_j - w, \bar{\mu} + w]$     ▷ Compute confidence interval for reward at current price
    if $L_j > L_{\max}$ then                                ▷ Keep track of the highest $L_j$
        $L_{\max} \leftarrow L_j$
    if $U_j < L_{\max}$ then                                ▷ Exploration phase ends
        $h \leftarrow j$                                     ▷ Define the *halting price*
        Break
Select price $x_h$ in all future rounds.                     ▷ Exploitation phase.
```

Theorem 25 **(Upper Bound for Infinite-Dimensional Family)**. *For any $s \geq 2$, the Uniform Elimination Policy satisfies* $\mathrm{Reg}(\mathrm{UE}_{m,\Delta}, \mathcal{F}_s^U) = O(T^{\frac{2s+1}{3s+1}})$.

We complement the above theorem with a lower bound in terms of both s and T, that matches the upper bound in Theorem 25 for every $s \geq 2$.

Theorem 26 **(Lower Bound for s-Sensitive Family)**. *For any $s \geq 2$, there is a family $\mathcal{F}$ of s-sensitive unimodal revenue curves satisfying Assumptions (1)-(4) such that any markdown policy π satisfies* $\mathrm{Reg}(\pi, \mathcal{F}) = \Omega(T^{\frac{2s+1}{3s+1}})$.

Intuitively, this lower bound is caused by the following trade-off. On the one hand if we reduce the prices too fast, we may have overshot by a lot when we halt; on the other hand if the speed is too slow we may spend too much time at suboptimal prices, incurring a high regret.

This tight regret bound, $T^{(2s+1)/(3s+1)}$, highlights how sensitivity helps reduce the regret for markdown pricing. Interestingly, as s grows to infinity, the regret approaches $T^{2/3}$, matching the regret of the unconstrained pricing problem *without* any smoothness assumption.

IV.4. Upper Bounds

In this section, we prove the following tight regret bounds for the markdown version. To highlight the technical challenges, we first rephrase the known tight regret bound for non-markdown version.

Theorem 27 **(Broder and Rusmevichientong (2012))**. *For any zero-dimensional demand family $\mathcal{F}$, there is an algorithm with regret $O(\log T)$. Moreover, there exists a zero-dimensional demand family $\mathcal{F}$ on which any algorithm has regret $\Omega(\log T)$.*

They considered a simple policy that estimates the true parameter using maximum likelihood estimator (MLE), and then selects the optimal price of the estimated demand function. To bound the expected regret in round t, they showed that the *mean squared error* (MSE) of the estimated price is at most $1/t$, and hence the expected total regret is $\sum_{t=1}^{T} \frac{1}{t} \sim \log T$.

IV.4.1. Zero-Dimensional Family While Theorem 27 is established by bounding the Mean Square Error (MSE), due to the monotonicity constraint for markdown pricing, it no longer suffices to consider the *mean* error. Rather, we need an error bound which (i) holds with high probability, so that we can make *conservative* decision by selecting a price that is extremely unlikely to overshoot the optimal price, and (ii) is sufficiently low, so that the total regret is also low. The following lemma can be obtained as a direct consequence of Hoeffding's inequality (Theorem 19).

Lemma 29. *Let $Z_1, .., Z_m$ be a i.i.d. samples from a distribution D with subgaussian norm C. Let $\mathcal{B}$ be the event that $\left|\mathbb{E}[D] - \frac{1}{m}\sum_{j=1} Z_j\right| \leq 2C \cdot \sqrt{\frac{\log T}{m}}$, then $\mathbb{P}[\overline{\mathcal{B}}] \leq T^{-2}$.*

Proof. By the Hoeffding inequality (Theorem 19), we have

$$\mathbb{P}\left[\left|\mathbb{E}[D]-\frac{1}{m}\sum_{j=1}^{m} Z_j\right| > 2C\sqrt{\frac{\log T}{m}}\right] \le \exp\left(-\frac{(2C\sqrt{t_j\log T})^2}{2t_j\cdot C^2}\right) = T^{-2}. \quad \square$$

Define $\mathcal{E}_j$ to be the event that $\left|D(p_j;\theta^*)-\bar{d}_j\right| \le 2C_{sg}\sqrt{\frac{\log T}{t_j}}$, where we recall that C_{sg} is the upper bound on the subgaussian norm of the demand distributions at any price, as formalized in Assumption 2. Consider $\mathcal{E} = \bigcap_{j=1}^m \mathcal{E}_j$. Note that D_j for $j = t^{(j-1)}+1,...,t^{(j)}$ are i.i.d. samples from a subgaussian distribution with mean $D(p_j;\theta^*)$, and that Assumption 2 the sugaussian norm of this distribution is at most C_{sg}. Thus by Lemma 29, we have $\mathbb{P}[\overline{\mathcal{E}_j}] \le T^{-2}$. By the union bound, we have

$$\mathbb{P}[\mathcal{E}] \ge 1 - T^{-2}\cdot\log T \ge 1-T^{-1}.$$

Since the expected regret per round is at most $[0,1]$, we can condition on $\mathcal{E}$ by losing only an $O(1)$ term in the regret.

Conditional on $\mathcal{E}$, the true parameter θ^* is contained in the confidence interval $I_j = [\bar{d}_j - w_j, \bar{d}_j + w_j]$ where $w_j = 2C_{sg}\sqrt{\frac{\log T}{t_j}}$, so the next selected price $p_{j+1} = \max\{p^*(\theta) : \theta \in I_j\}$ satisfies $p_j \ge p^*(\theta^*)$, i.e. our policy does not overshoot the optimal price.

We next explain why the estimated price is close to $p^*(\theta^*)$. Since Φ is a robust parametrization, by definition we have

$$\begin{aligned}\|\hat{\theta}_j-\theta^*\|_2 &= \|\Phi_{p_j}^{-1}(\bar{d}_j) - \Phi_{p_j}^{-1}\left(\Phi_{p_j}(\theta^*)\right)\| \\ &\le C_2\cdot\|\bar{d}_j - \Phi_{p_j}(\theta^*)\| \\ &= C_2\cdot\|\bar{d}_j - D(p_j;\theta^*)\| \le 2C_2\cdot C_{sg}\sqrt{\frac{\log T}{t_j}}.\end{aligned}$$

Moreover, by Assumption 5, the mapping p^* is C^*-Lipschitz for some constant $C^*>0$, so the price p_{j+1} selected in the $(j+1)$-st phase satisfies

$$|p_{j+1}-p^*(\theta^*)| \le C^*\|\hat{\theta}_j - \theta^*\|_2 \le 2C_2\cdot C^*\cdot C_{sg}\sqrt{\frac{\log T}{t_j}}.$$

Since the length of phase $j+1$ is t_{j+1}, the regret incurred in this phase is at most $C_s\left(2C^*\cdot C_2\cdot C_{sg}\sqrt{\frac{\log T}{t_j}}\right)^s\cdot t_{j+1}$ in expectation. Note that there are in total $K \le \log T - \log\log T$ phases, so we can bound the cumulative regret as

$$\begin{aligned}\mathrm{Reg}(\mathrm{CM},\mathcal{F}) &\le \sum_{j=1}^{K} C_s\left(2C^*\cdot C_2\cdot C_{sg}\sqrt{\frac{\log T}{t_j}}\right)^s\cdot t_{j+1} \\ &= C_s\left(2C^*\cdot C_2\cdot C_{sg}\sqrt{\log T}\right)^s\cdot\sum_{j=0}^{K}\frac{t_{j+1}}{t_j^{s/2}} \qquad (34)\end{aligned}$$

We substitute t_j with $\lceil 2^j \log T \rceil$ and simplify the above for $s = 2$ and $s > 2$ separately. When $s = 2$,

$$\begin{aligned}
(34) &= C_s \left(2C^* \cdot C_2 \cdot C_{sg}\sqrt{\log T}\right)^2 \cdot \sum_{j=0}^{K} \frac{t_{j+1}}{t_j} \\
&= C_s \left(2C^* \cdot C_2 \cdot C_{sg}\right)^2 \log T \cdot (\log T - \log\log T) \\
&= O(\log^2 T).
\end{aligned}$$

Now suppose $s > 2$. Then,

$$\begin{aligned}
(34) &= C_s \left(2C^* \cdot C_2 \cdot C_{sg}\sqrt{\log T}\right)^s \cdot \sum_{j=0}^{K} \frac{2^{j+1}\log T}{2^{j\cdot\frac{s}{2}} \log^{s/2} T} \\
&\le C_s \left(2C^* \cdot C_2 \cdot C_{sg}\right)^s \cdot \log T \cdot \sum_{j=0}^{K} 2^{(1-\frac{s}{2})j+1} \\
&\le 2C_s \cdot \left(2C_2 \cdot C^* \cdot C_{sg}\right)^s \cdot \log T \cdot \int_0^K 2^{(1-\frac{s}{2})x} dx \\
&= 2C_s \cdot \left(2C^* \cdot C_2 \cdot C_{sg}\right)^s \cdot \log T \cdot \frac{2}{(s-2)\cdot \ln 2} \\
&= O(\log T).
\end{aligned}$$

Theorem 21 follows by combining the analyses for $s = 2$ and $s > 2$. □

IV.4.2. Finite-Dimensional Family In this section we first analyze the regret of the ICM policy and prove Theorem 23, and then complement this upper bound with an almost matching lower bound. Recall that the ICM policy is specified by two types of parameters: the gap h between neighboring sampling prices in each phase, and the number T_j of rounds to stay at each sampling price in phase j. To prove Theorem 23, we first present the following upper bound on the regret of ICM for arbitrary choice of parameters h and T_j's, and then optimize the choice of parameters (up to polylogarithmic factors in T) by solving a linear program.

Proposition 11. *Let $\mathcal{F}$ be a d-dimensional, s-sensitive ($s \ge 2$) family of demand functions. Suppose $h > 0$ and $0 < T_1 < ... < T_m$ where $T_m = o(T)$. Denote* $\mathrm{ICM} = \mathrm{ICM}(T_1, ..., T_m, h)$. *Then,*

$$\mathrm{Reg}(\mathrm{ICM}, \mathcal{F}) \le T_1 + C_s \left(2C^* C_{sg} h^{-d} \sqrt{C_5 d \log T}\right)^s \cdot \left(\sum_{j=1}^{m-1} T_{j-1}^{-s/2} \cdot T_j + T_m^{-s/2} \cdot T\right) + C_s (mdh)^s T.$$

We briefly explain the intuition behind the above result before proceeding with finding the optimal parameters. As the name suggests, the Iterative Cautious Myopic policy iteratively computes a confidence interval $[L_j, U_j]$ around the true optimal price, and conservatively moves to the right endpoint of this interval. As a *simplistic* view, in phase j (assuming it ever takes place) the estimation error is $\sim h^{-d}T_{j-1}^{-1/2}$, and by definition of s-sensitivity, the regret incurred in phase j is $\sim (h^{-d}T_{j-1}^{-1/2})^s T_j$.

To understand the final term, observe that when h is sufficiently small compared to $U_j - L_j$, there is little risk of *overshooting* at the right endpoint U_j. However, when one selects larger h (for faster learning rate), it may happen that the last sample price $p_j - dh$ in this phase overshoots the optimal price, thereby incurring a regret term, as captured by the last term in the above bound.

Nonetheless, the actual proof involves carefully analyzing each of the three events (good, dangerous and overshooting) that can possibly occur at the end of each phase, as formally defined in Algorithm 10. Informally, each of these three events corresponds to the scenario where the price at the end of this phase lies (1) on the right, (2) inside, or (3) on the left of the confidence interval of the estimated optimal price.

LEMMA 30. *For each phase $j = 1, ..., m$, let $\mathcal{E}_j$ be the event that $p^*(\theta^*) \in [L_j, U_j]$. Then, $\mathbb{P}(\mathcal{E}_j) \ge 1 - dT^{-2}$.*

Proof. By the Hoeffding inequality (Theorem 19) and the subgaussian assumption (Assumption 5), for each $k = 0, ..., d$, it holds with probability at least $1 - T^{-2}$ that

$$|D(p_i - kh; \theta^*) - \bar{d}| \le 2C_{sg}\sqrt{\frac{\log T}{T_j}}.$$

For simplicity we write $\Phi = \Phi_{p_i, p_i - h, ..., p_i - dh}$ and $\bar{\mathbf{d}} = (\bar{d}_0, ..., \bar{d}_d)$. Since for any $v \in \mathbb{R}^d$ it holds $\|v\|_2 \le \sqrt{d} \cdot \|v\|_\infty$, it holds with probability $1 - (d+1)T^{-2}$ that

$$\|\Phi(\theta^*) - \bar{\mathbf{d}}\|_2 \le 2C_{sg}\sqrt{\frac{\log T}{T_j}} \cdot \sqrt{d}.$$

Thus by definition of dimension, for sufficiently large T_j (hence sufficiently small $\|\Phi(\theta^*) - \bar{\mathbf{d}}\|_2$),

$$\begin{aligned}\|\theta^* - \hat{\theta}\|_2 &= \|\Phi^{-1}(\bar{\mathbf{d}}) - \Phi^{-1}(\Phi(\theta^*))\|_2 \\ &\le C_2 h^{-d} \cdot \|\Phi(\theta^*) - \bar{\mathbf{d}}\|_2 \\ &\le C_2 h^{-d} \cdot C_{sg} 2\sqrt{\frac{\log T}{T_j}} \cdot \sqrt{d} = w_j,\end{aligned}$$

and $\theta^* \in [L_j, U_j]$ follows immediately from the definition of L_j and U_j. □

Proof of Proposition 11. We first show that with high probability, our confidence interval forms a nested sequence of intervals containing the true parameter θ^*. Recall that

$$L_j = \min\{p^*(\theta) : \|\theta - p^*(\hat{\theta})\|_2 \le w_j\} \quad \text{and} \quad U_j = \max\{p^*(\theta) : \|\theta - p^*(\hat{\theta})\|_2 \le w_j\}.$$

This lemma immediately implies a (high-probability) upper bound for the estimation error of the optimal price. Recall that $p_j = \max\{p^*(\theta) : \|\theta - p^*(\hat{\theta})\|_2 \le w_j\}$. By Lemma 30 and Assumption 5, we deduce that conditional on $\mathcal{E}_j$, for any $p \in [L_j, U_j]$ it holds

$$|p - p^*(\theta^*)| \le C^* \|\theta^* - \hat{\theta}\|_2 \le C^* w_j.$$

We will repeatedly apply this bound in the following regret analysis.

Proof of Proposition 11. By Lemma 30,

$$\mathbb{P}(\bigcup_{i=1}^{m} \overline{\mathcal{E}_i}) \le \sum_{i=1}^{m} \mathbb{P}(\overline{\mathcal{E}_i}) \le dT^{-2} \cdot m \le T^{-1}.$$

Thus, we may subsequently assume $\bigcap_{i=1}^m \mathcal{E}_i$ occurs by losing only an $O\left(\frac{1}{T}\right)$-factor in regret.

We split our proof into two cases depending on whether the overshooting event ever occurs in any phase.

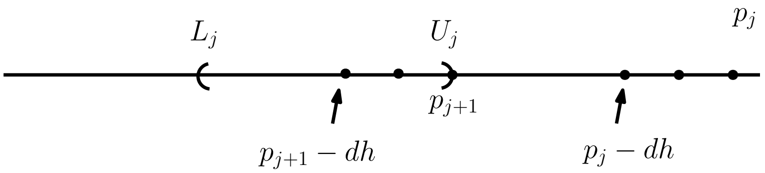

Figure 6 Illustration of case 2.

Case (1). Suppose the overshooting event never occurs, i.e. in each $j = 1, ..., m$, we always have $p_j - dh \ge L_j \ge L_{j-1}$. Since $p_j \le U_{j-1}$, we deduce that $p_j - kh \in [L_{j-1}, U_{j-1}]$ for all $k = 0, .., d$. On the other hand, since we have conditioned on $\bigcup_{i=1}^m \overline{\mathcal{E}_i}$, we have $p^*(\theta^*) \in [L_{j-1}, U_{j-1}]$, hence $|(p_j - kh) - p^*| \le U_{j-1} - L_{j-1} \le C^* w_j$ for $0 \le k \le d$. Thus the regret incurred in this phase is at most $C_s \cdot (U_{j-1} - L_{j-1})^s T_j \le C_s (C^* w_j)^s T_j$. Similarly, since the exploitation price $\tilde{p} := p_m - dh$ satisfies $|\tilde{p} - p^*(\theta^*)| \le C^* w_m$, the expected regret per round in the exploitation phase is at most $(C^* w_m)^s$. Therefore, we may bound the cumulative regret as

$$\begin{aligned}\mathrm{Reg}(\mathrm{ICM}, \mathcal{F}) &\le T_1 + C_s (C^* w_1)^s T_2 + ... + C_s (C^* w_{m-1})^s T_m + C_s (C^* w_m)^s T \\ &\le T_1 + \sum_{j=2}^{m} C_s \left(C^* \cdot 2C_{sg} h^{-d} \sqrt{\frac{C_5 d \log T}{T_{j-1}}} \right)^s T_j + C_s \left(C^* \cdot 2C_{sg} h^{-d} \sqrt{\frac{C_5 d \log T}{T_m}} \right)^s T \\ &= T_1 + C_s \left(2C^* C_{sg} h^{-d} \sqrt{C_5 d \log T} \right)^s \cdot \left(\sum_{j=1}^{m-1} T_{j-1}^{-s/2} \cdot T_j + T_m^{-s/2} \cdot T \right).\end{aligned}$$

Case (2). Now suppose the overshooting event first occurs in some phase ℓ where $1 \le \ell \le m$, formally

$$\ell = \min\{s : p_s - dh < L_j\}.$$

As in case (1), the expected regret in phase $j = 1, ..., \ell - 1$ can be bounded by $C_s \cdot (C^* w_{j-1})^s T_j$. Thus it remains to bound the expected regret in the ℓ-th and the exploitation phase as $\tilde{O}\left((mdh)^s T\right)$. Suppose the last phase that good event occurred is phase j (as illustrated in Fig 6). There are two sub-cases.

i) Suppose $j = \ell - 1$. Then, by definition of good event, we have $p_j - dh \ge U_j$. Thus, the ICM policy sets the next price to be $p_{j+1} = U_j$. Since $p^*(\theta^*) \in [L_j, U_j]$ and $p_\ell = p_{j+1} = U_j$, the exploitation price $p_\ell - dh$ satisfies

$$|p_\ell - dh - p^*(\theta^*)| \le |p_\ell - dh - U_j| = |p_\ell - dh - p_\ell| = dh.$$

Thus, the future regret is at most $C_s (dh)^s T$.

ii) Now suppose $j \le \ell - 2$. Then, the dangerous event must have occurred in phases $j+1, j+2, ... \ell - 1$, so

$$p_{j+s+1} = p_{j+s} - dh, \quad \forall s = 1, ..., \ell - j - 1.$$

In particular,

$$p_\ell = p_{j+1} - (\ell - j - 1) \cdot dh.$$

On the other side, by definition of the overshooting event, it holds

$$p_\ell - dh \le L_{j+1} \le p^*(\theta^*) \le U_{j+1} \le U_j = p_{j+1},$$

i.e. $p_\ell - dh \le p^*(\theta^*) \le p_{j+1}$. Thus,

$$|p_\ell - dh - p^*(\theta^*)| \le (\ell - j - 1)\, dh.$$

Therefore, the regret in the exploitation phase is bounded by $C_s |p_\ell - dh - p^*|^s T \le C_s (mdh)^s T$. The proof completes by combining the analyses for the above cases. □

We now determine the parameters to minimize the upper bound in Proposition 11.

Proof of Theorem 23. Write $T_i = T^{z_i}$, $h = T^{-y}$. Then for any $j \le m - 1$,

$$(h^{-d} T_j^{-1/2})^s T_{j+1} = T^{sdy - \frac{s}{2} z_j + z_{j+1}}.$$

To find the optimal parameters, consider

$$\begin{aligned}
\mathrm{LP}(d): \quad & \min_{x,y,z} & T^{x} & & \\
& \text{subject to} & T^{z_1} &\leq T^{x}, & \text{Regret in phase 1} \\
& & T^{2sdy+z_2-\frac{s}{2}z_1} &\leq T^{x}, & \text{Regret in phase 2} \\
& & & \ldots & \\
& & T^{sdy+1-\frac{s}{2}z_m} &\leq T^{x}, & \text{Regret in the exploitation phase} \\
& & T^{1-sy} &\leq T^{x}, & \text{Regret for overshooting} \\
& & x, y, z \geq 0, & z \leq 1 &
\end{aligned}$$

Taking logarithm with base T on both sides, the above becomes

$$\begin{aligned}
\min_{x,y,z} \quad & x \\
\text{s.t.} \quad & \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & \ldots & 0 \\ -1 & sd & -\frac{s}{2} & 1 & 0 & 0 & \ldots & 0 \\ -1 & sd & 0 & -\frac{s}{2} & 1 & 0 & \ldots & 0 \\ -1 & sd & 0 & 0 & -\frac{s}{2} & 1 & \ldots & 0 \\ & & & \ldots\ldots & & & & \\ -1 & sd & 0 & 0 & 0 & \ldots & -\frac{s}{2} & 1 \\ -1 & sd & 0 & 0 & 0 & 0 & \ldots & -\frac{s}{2} \\ -1 & -s & 0 & 0 & 0 & 0 & \ldots & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z_1 \\ z_2 \\ \vdots \\ z_{m-1} \\ z_m \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ -1 \\ -1 \end{bmatrix} \\
& x, y, z \geq 0, \quad z \leq 1
\end{aligned}$$

Note that the above LP consists of $m+2$ variables and $m+2$ inequality constraints, so the minimum is attained when all inequalities become identities. In this case, we have

$$z_1 = x \tag{35}$$

$$\begin{aligned}
z_2 - \frac{s}{2}z_1 &= x - sdy \\
z_3 - \frac{s}{2}z_2 &= x - sdy \\
&\ldots \\
z_m - \frac{s}{2}z_{m-1} &= x - sdy \\
1 - \frac{s}{2}z_m &= x - sdy
\end{aligned}$$

$$1 - sy = x \tag{36}$$

By telescoping sum, we have

$$1 - \left(\frac{s}{2}\right)^m z_1 = \left(1 + \frac{s}{2} + \ldots + \left(\frac{s}{2}\right)^{m-1}\right) \cdot (x - dsy).$$

Combining the above with (35) and (36), we have

$$1+\left(1+\frac{s}{2}+...+\left(\frac{s}{2}\right)^{m-1}\right)d(1-x)=\left(1+\frac{s}{2}+...+\left(\frac{s}{2}\right)^{m}\right)x.$$

Rearranging, we obtain

$$x=\frac{1+\left(1+\frac{s}{2}+...+(\frac{s}{2})^{m-1}\right)d}{\left(1+\frac{s}{2}+...+(\frac{s}{2})^{m-1}\right)\cdot(d+1)+\left(\frac{s}{2}\right)^{m}}.$$

In particular, for $s=2$, the above becomes

$$x=\frac{md+1}{m(d+1)+1}=\frac{d}{d+1}+\frac{1}{m(d+1)^2}.$$

Choosing $m=\log T$, we have $T^x=\tilde{O}(T^{\frac{d}{d+1}})$. □

IV.4.3. Infinite- Dimensional Family In this section we first present a general regret upper bound for policy $\mathrm{UE}_{\Delta,w}$, which immediately implies Theorem 25. To this aim, we need to introduce another constant η, as motivated by the the following result. For notational convenience, we abbreviate $\frac{\partial^k}{\partial x^k}R(x;\theta)$ as $R^{(k)}(x;\theta)$ for any $k\geq 0$.

Lemma 31. *Let $\mathcal{F}=\{R(x,\theta):\theta\in\Theta\}$ be a family of s-sensitive reward functions. Then, there exists a constant $\eta>0$ such that for any $\theta\in\Theta$ and $x\in[p^*(\theta)-\eta,p^*(\theta)]$, it holds $R^{(s)}(x;\theta)<0$ and*

$$2R^{(s)}(p^*(\theta);\theta)\leq R^{(s)}(x;\theta)\leq\frac{1}{2}R^{(s)}(p^*(\theta);\theta).$$

Proof. First consider any fixed $\theta\in\Theta$. By definition of sensitivity, we have $R^{(1)}(p^*(\theta),\theta)=...=R^{(s-1)}(p^*(\theta),\theta)=0$ and $R^{(s)}(p^*(\theta),\theta)<0$. Define

$$g(\theta)=\sup\left\{\gamma\geq 0\mid 2R^{(s)}(p^*(\theta);\theta)\leq R^{(s)}(x;\theta)\leq\frac{1}{2}R^{(s)}(p^*(\theta);\theta),\quad\forall x\in[p^*(\theta)-\gamma,p^*(\theta)]\right\}.$$

By continuity of $R^{(s)}$ in x, we have $g(\theta)>0$ for any $\theta\in\Theta$. We complete the proof by showing that $\eta:=\sup_{\theta\in\Theta}g(\theta)>0$. Recall that Θ is compact, and $R^{(s)}$ is continuous in θ, we know that η can be attained, i.e., there exists some $\theta\in\Theta$ with $g(\theta)=\eta$. Moreover, note that for any θ we have $g(\theta)>0$, therefore $\eta>0$, and the proof completes. □

We are now ready to state the main result in this section. Note that by choosing $\Delta=T^{-1/(3s+1)}$ and $w=T^{-2/(3s+1)}$, we immediately obtain Theorem 25.

Proposition 12 **(Upper Bound)**. *Let $\mathcal{F}$ be any s-sensitive family for some $s\geq 2$. Suppose $\Delta\leq\frac{C_s}{8s!C^{(1)}}\eta^s$ and $m\geq 4$, then*

$$\mathrm{Reg}\left(\mathrm{UE}_{\Delta,m},\mathcal{F}\right)=O\left(\Delta^{-1}w^{-2}\log T+(w+\Delta^s)T\right)$$

where we recall that $w=2C_{sg}\sqrt{\frac{\log T}{m}}$.

Our analysis proceeds by conditioning on the following the notion of clean event, which occurs with high probability as we will show soon.

Definition 29 (Clean event). Let $\mathcal{E}_j$ be the event that $\left|R(x_j) - \bar{\mu}_j\right| \le 2C_{sg}\sqrt{\frac{\log T}{m}}$, and $\mathcal{E} = \bigcap_{j=1}^{\lceil \Delta^{-1} \rceil} \mathcal{E}_j$.

Note that by our choice of L_j, U_j, we know that $\mathcal{E}$ is simply the event that $R(x_j) \in [L_j, U_j]$ for all $1 \le j \le \Delta^{-1}$. We next show that $\mathcal{E}$ occurs with high probability, and hence we may perform the analysis conditional on $\mathcal{E}$.

Lemma 32. $\mathbb{P}(\overline{\mathcal{E}}) \le T^{-1}$.

Proof. Let R be the true reward function. Recall that Z_i^j for $i = t^{(j-1)} + 1, ..., t^{(j)}$ are i.i.d. samples from a subgaussian distribution with mean $R(x_j)$, and that Assumption 2 the sugaussian norm of this distribution is at most C_{sg}. Thus by Lemma 29, we have $\mathbb{P}[\overline{\mathcal{E}_j}] \le T^{-2}$. By the union bound, we have $\mathbb{P}[\mathcal{E}] \ge 1 - T^{-2} \cdot \log T \ge 1 - T^{-1}$. □

In the rest of this section we will fix a true reward function $R(x;\theta)$ and write $x^* = p^*(\theta)$ and $R(x) = R(x;\theta)$.

Definition 30. Define x_ℓ be the closest sample price to x^*, i.e.

$$\ell := \underset{0 \le j \le \Delta^{-1}}{\arg\min} \{|x_j - x^*|\}.$$

We first show that conditional on $\mathcal{E}$, the policy will stop reducing the price and enter the exploitation phase before reaching $x^* - \eta$.

Lemma 33. *Suppose $\mathcal{E}$ occurs. For any $m \ge \left(\frac{4s! \cdot 8 \cdot C_{sg}}{C_s}\right)^2 \cdot \eta^{-2s} \log T$ and $\Delta \le \frac{C_s}{8s! C^{(1)}} \eta^s$, we have $x_h \ge x^* - \eta$.*

Proof. Recall that x is said to be a sample price if $x = 1 - j\Delta$ for some integer j. Consider the smallest sample price $\tilde{x}$ above $x^* - \eta$, then $|x^* - \eta - \tilde{x}| \le \Delta$. By Assumption 4, the first derivatives are bounded by $C^{(1)}$ and hence R is $C^{(1)}$-Lipschitz, so

$$|R(\tilde{x}) - R(x^* - \eta)| \le C^{(1)} |x^* - \eta - \tilde{x}| \le C^{(1)}\Delta \quad \text{and} \quad |R(x_\ell) - R(x^*)| \le C^{(1)}\Delta.$$

Moreover, by Theorem 20, and since $R^{(1)}(x^*) = ... = R^{(s-1)}(x^*) = 0$,

$$|R(x^* - \eta) - R(x^*)| = \left| \frac{R^{(s)}(\xi)}{s!} \eta^s \right| \tag{37}$$

for some $\xi \in (x^* - \eta, x^*)$. By Lemma 31, $|R^{(s)}(\xi)| \ge \frac{1}{2} \cdot |R^{(s)}(x^*)|$, so

$$|R(x^* - \eta) - R(x^*)| \ge \frac{|R^{(s)}(x^*)|}{2s!} \eta^s. \tag{38}$$

By combining the inequalities (37) and (38), we have

$$R(\tilde{x}) \le R(x_\ell) - \left(\frac{|R^{(s)}(x^*)|}{2s!} \eta^s - 2C^{(1)}\Delta \right).$$

Recall that $|R^{(s)}(x^*)| \ge C_s$, so for any $\Delta \le \frac{C_s}{8s! C^{(1)}} \eta^s$, we have

$$\frac{|R^{(s)}(x^*)|}{2s!} \eta^s - 2C^{(1)}\Delta \ge \frac{|R^{(s)}(x^*)|}{4s!} \eta^s. \tag{39}$$

Hence, suppose $m \ge \left(\frac{4s! \cdot 8 \cdot C_{sg}}{C_s} \right)^2 \cdot \eta^{-2s} \log T$, then $4w \le \frac{C_s}{4s!} \eta^s \le \frac{|R^{(s)}(x^*)|}{4s!} \eta^s$, and by (39)

$$R(\tilde{x}) < R(x_\ell) - 4w. \tag{40}$$

Since $\mathcal{E}$ occurs, we have $|U(\tilde{x}) - R(\tilde{x})| \le w$ and $|L(x_\ell) - R(x_\ell)| \le w$. Combining with (40), we obtain $U(\tilde{x}) < L(x_\ell)$, and thus the halting criterion is satisfied at $\tilde{x}$, so $x_h \ge x^* - \eta$. □

LEMMA 34. *Let* $i := \arg\max_{0 \le j \le \Delta^{-1}} \{L_j\}$. *Then for sufficiently small* Δ, *for any* $k \ge 3$ *it holds that*

$$|R(x_{\ell+k}) - R(x_i)| \ge \frac{C_s}{2^s s!} (k\Delta)^s - 4w.$$

Proof. The proof can be split into showing the following inequalities:

1. $|R(x^*) - R(x_\ell)| \le \frac{C_s}{s!} \Delta^s$,
2. $|R(x_i) - R(x_\ell)| \le \max\{4w, \frac{2C_s}{s!} \Delta^s\}$,
3. $|R(x_{k+\ell}) - R(x^*)| \ge \frac{C_s}{s!} \cdot ((k-1)\,\Delta)^s$ for $k \ge 3$.

Assuming these three inequalities are true, then by triangle inequality,

$$\begin{aligned} |R(x_{\ell+k}) - R(x_i)| &\ge |R(x_{\ell+k}) - R(x^*)| - |R(x^*) - R(x_\ell)| - |R(x_i) - R(x_\ell)| \\ &\ge \frac{C}{s!} \cdot ((k-1)\,\Delta)^s - \frac{C_s}{s!} \Delta^s - 4w + \frac{C_s}{s!} \Delta^s \\ &\ge \frac{C_s}{s!} \Delta^s \left((k-1)^s - 1 \right) - 4w \\ &\ge \frac{C_s}{2^s s!} (k\Delta)^s - 4w. \end{aligned}$$

We now prove each of these three inequalities respectively.

Step 1. Applying Theorem 20 on the reward function R by setting $x = x^*$ and $x' = x_\ell$, we deduce that there exists $\xi \in [x_\ell, x^*]$ with

$$\begin{aligned} R(x_\ell) &= R(x^*) + \frac{R'(x^*)}{1!} \cdot (x_\ell - x^*) + \ldots + \frac{R^{(s-1)}(x^*)}{(s-1)!} (x_\ell - x^*)^{s-1} + \frac{R^{(s)}(\xi)}{s!} \cdot (x_\ell - x^*)^s \\ &= R(x^*) + \frac{1}{s!} R^{(s)}(\xi) \cdot (x_\ell - x^*)^s. \end{aligned}$$

Thus, when $|x_\ell - x^*|$ is sufficiently small, we have

$$|R(x^*) - R(x_\ell)| \le \frac{2C_s}{s!}|x_\ell - x^*|^s \le \frac{2C_s}{s!}\Delta^s,$$

where we used $|x_\ell - x^*| \le \Delta$.

Step 2. This is direct consequence of Step 1. In fact,

$$R(x_\ell) \ge R(x^*) - \frac{2C_s}{s!}\Delta^s \ge R(x_i) - \frac{2C_s}{s!}\Delta^s.$$

On the other side, by definition of x_i and event $\mathcal{E}$, we have

$$R(x_i) \ge R(x_\ell) - 4w.$$

Combining, we have $|R(x_i) - R(x_\ell)| \le \max\{4w, \frac{2C_s}{s!}\Delta^s\}$.

Step 3. Applying Theorem 20 on $x' = x_{k_\ell}$, we deduce that there exists $\xi \in [x_{k+\ell}, x^*]$ with

$$\begin{aligned} R(x_{k+\ell}) &= R(x^*) + \frac{R'(x^*)}{1!} \cdot (x_{k+\ell} - x^*) + ... + \frac{R^{(s-1)}(x^*)}{(s-1)!}(x_{k+\ell} - x^*)^{s-1} + \frac{R^{(s)}(\xi)}{s!} \cdot (x_{k+\ell} - x^*)^s \\ &= R(x^*) + \frac{1}{s!}R^{(s)}(\xi) \cdot (x_{k+\ell} - x^*)^s. \end{aligned} \tag{41}$$

Applying Lemma 31, we have $R^{(s)}(\xi) \ge C_s$, thus by (41)

$$|R(x_{k+\ell}) - R(x^*)| \ge \frac{C_s}{s!} \cdot |x_{k+\ell} - x^*|^s$$

Note that by definition of x_ℓ, it holds, $|x^* - x_\ell| \le \Delta$, and thus

$$|x^* - x_{k+\ell}| \ge |x_\ell - x_{k+\ell}| - |x^* - x_\ell| \ge (k-1)\Delta,$$

hence

$$|R(x_{k+\ell}) - R(x^*)| \ge \frac{C_s}{s!} \cdot ((k-1)\,\Delta)^s\,. \qquad \square$$

The crux of our proof lies in analyzing the regret in the exploitation phase. To this aim, we use the above lemma to bound the per-round regret in the exploitation phase, formally stated below.

Lemma 35. *Suppose $\mathcal{E}$ occurs, then the halting price x_h satisfies*

$$R(x_h) - R(x^*) \le 6 \cdot 3^s s! \cdot C_s \cdot w + \max\{3^s, C_s\} \cdot \Delta^s.$$

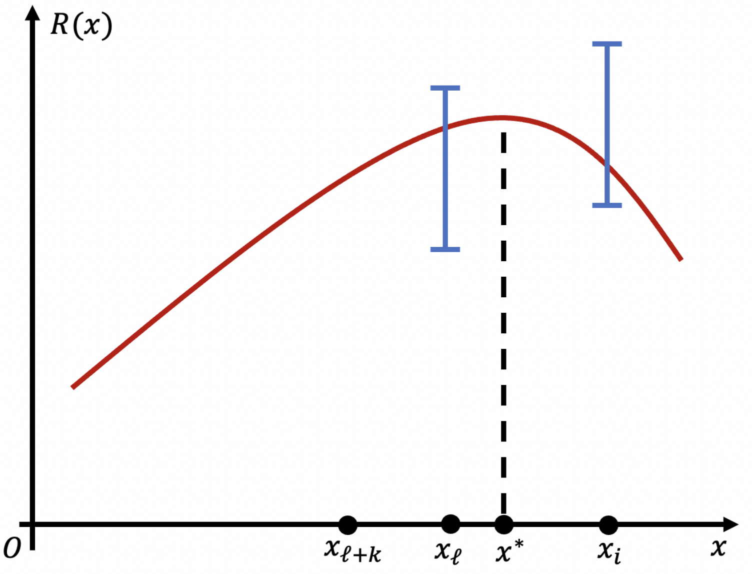

Figure 7 **Illustration of Lemma 35**

Proof. Consider any true reward function $R \in \mathcal{F}$.

Case 1. Suppose $h \geq \ell - 2$, i.e. $x_h \geq x_\ell - 2\Delta$. Since $|x_\ell - x^*| \leq \Delta$, we have

$$|x_h - x^*| \leq |x_\ell - x_h| + |x^* - x_\ell| \leq 2\Delta + \Delta = 3\Delta.$$

Thus by definition of sensitivity, when $|x_h - x^*| \leq \eta$ it holds

$$|R(x^*) - R(x_h)| \leq C_s \cdot |x_h - x^*|^s \leq C_s \cdot 3^s \Delta^s.$$

Case 2: Now suppose $h \leq \ell - 3$, i.e. $x_h \leq x_\ell - 3\Delta$. Let $k = h - \ell - 1$, so that $x_{\ell+k}$ is the last sample price that the UE policy selected before halting at x_h. Then by definition of x_h, the halting criterion is *not* satisfied at the $x_{\ell+k}$, i.e. $[L(x_i), U(x_i)] \cap [L(x_{\ell+k}), U(x_{\ell+k})] \neq \emptyset$, so

$$|R(x_i) - R(x_{\ell+k})| \leq 4w.$$

Combining with Lemma 34, we have

$$4w \geq |R(x_i) - R(x_{\ell+k})| \geq \frac{C_s}{2^s s!}(k\Delta)^s - 2w,$$

i.e.

$$(k\Delta)^s \leq \frac{2^s s! \cdot 6w}{C_s}. \tag{42}$$

Note that $|x_h - x^*| \le (k+1)\Delta$, we obtain

$$\begin{aligned}
|R(x_h) - R(x^*)| &\le C_s \cdot ((k+1)\Delta)^s && \text{by sensitivity}\\
&\le C_s \cdot \left(\frac{3}{2}k\Delta\right)^s && \text{since } k = h - \ell - 1 \ge 2\\
&\le \frac{6w \cdot C_s \cdot 3^s s!}{C_s} = 6 \cdot 3^s s! \cdot C_s \cdot w && \text{by (42)}
\end{aligned}$$

and the proof is complete. □

Proof of Proposition 12. Fix any $R \in \mathcal{F}$. Suppose $\mathcal{E}$ does not occur, then the regret is at most T. Suppose $\mathcal{E}$ occurs, then by Lemma 35, the regret incurred in the exploitation phase is bounded by $\left(\frac{6 \cdot C_s \cdot 3^s s!}{C_6} \cdot w + \max\{3^s, C_s\} \cdot \Delta^s\right) T$.

On the other side, recall that each sample price is selected for at most m times, so the cumulative regret incurred in the exploration phase is bounded by mT. Moreover, there are at most $\lceil \Delta^{-1} \rceil$ sample prices. Recalling that $w = 2C_{sg}\sqrt{\frac{\log T}{m}}$, i.e. $m = 4C_{sg}^2 w^{-2} \log T$, we can bound the total regret as

$$\begin{aligned}
\text{Reg}(\text{UE}_{\Delta,w}, R) &\le \mathbb{P}[\overline{\mathcal{E}}] \cdot T + \mathbb{P}[\mathcal{E}] \cdot \left(4C_{sg}^2 w^{-2}\Delta^{-1}\log T + \left(\frac{6 \cdot C_s \cdot 3^s s!}{C_6} \cdot w + \max\{3^s, C_s\} \cdot \Delta^s\right) \cdot T\right)\\
&\le T^{-1} \cdot T + \left(4C_{sg}^2 w^{-2}\Delta^{-1}\log T + \left(\frac{6 \cdot C_s \cdot 3^s s!}{C_6} \cdot w + \max\{3^s, C_s\} \cdot \Delta^s\right)\right) \cdot T\\
&= O\left(\Delta^{-1} w^{-2} \log T + (\Delta^s + w)T\right),
\end{aligned}$$

and Proposition 12 follows. □

IV.5. Lower Bounds

IV.5.1. Preliminaries We now turn to proving our lower bound, which establishes minimax optimality. Our proof considers *Bernoulli* reward distribution at each price and employs the following alternate view of a *policy* as binary decision trees, which we will make precise in this section.

Definition 31 (Prefix). Let $\{0,1\}^* = \bigcup_{n=1}^\infty \{0,1\}^n \cup \{\text{null}\}$ be the set of all finite-length binary vectors, where *null* denotes the empty binary vector. For any $v \in \{0,1\}^*$ and $k \in \mathbb{Z}$, the *k-prefix* of v is defined as $v^k = (v_1, ..., v_k)$.

We will consider probability spaces on sets containing the prefixes of every element.

Definition 32 (Downward Closed Sets). For any $v, w \in \{0,1\}^*$, we define $w \prec v$ if there exists $k \in \mathbb{Z}$ such that $v^k = w$. A set $\Omega \in \{0,1\}^*$ is *downward closed*, if for any $v \in \Omega$ and $w \prec v$, we have $w \in \Omega$.

A decision tree is specified by a downward closed set equipped with a real-valued function.

Definition 33 (Decision Tree). A *binary decision tree* is a tuple (Ω, x) where $\Omega \subseteq \{0,1\}^*$ is downward closed and $x : \Omega \to \mathbb{R}$ is a mapping. Moreover, each $v \in \Omega$ is called a *node*.

Intuitively, for each node $v = (v_1, ..., v_k)$, the value $x(v)$ is just the *price* that the policy selects upon observing demands $v_1, ..., v_k$ at prices $x(v^1), ..., x(v^k)$. Recalling that we have normalized the price space to be $[0,1]$, so we will subsequently consider only decision trees (Ω, x) with $0 \le x(v) \le 1$ for all $v \in \Omega$. For notational convenience, we suppress the notation $x(v)$ simply as x_v.

We next introduce an equivalent definition of markdown policy, using the language of decision tree.

Definition 34 (Markdown Policy, Equivalent Definition). A *markdown policy* is a decision tree (Ω, x) such that it holds that $x(v^1) \ge x(v^2) \ge ... \ge x(v^k)$ for any $v = (v_1, ..., v_k) \in \Omega$. One may verify that this definition of markdown policy is indeed equivalent to the one given in Section IV.2. We next introduce some standard terminologies for decision trees, in case the reader is not familiar with graph theory.

Definition 35 (Decision Tree Basics). Let $\mathbb{A} = (\Omega, x)$ be a decision tree and $v, w \in \Omega$.

i). We say v is a *leaf* if there does not exist $w \in \Omega$ with $v \prec w$.

ii). The *depth* $d(v)$ of v is defined to be the length of binary vector v. Denote $L(\Omega) \subseteq \Omega$ the subset of all leaves. Each node in $\Omega \backslash L(\Omega)$ is called an *internal* node.

iii). We say w is an *ancestor* of v if $w \prec v$. If in addition, $d(v) = d(w) + 1$, then we say w is the *parent* of v and denote $w = par(v)$, and say v is a *child* of w.

iv). A decision tree is *binary* if every internal has exactly two children.

Given a binary decision tree, every reward function induces a natural probability measure over the leaves. In fact, consider a random walk from the root to a random leaf, where at each internal node v, the walk moves to each of the two children with probability $R(x_v)$ and $1 - R(x_v)$ respectively, corresponding to whether there is a unit demand in this round. We formally define this probability measure below.

Definition 36 (Probability Measure on Leaves). Let (Ω, x) be a decision tree and $R : [0,1] \to [0,1]$. Write $L = L(\Omega)$. For each $\ell = (\ell_1, ..., \ell_d) \in L$, define

$$p_R(\ell) = \prod_{j=1}^{d} R\left(x(\ell^j)\right)^{\ell_j} \cdot \left(1 - R\left(x(\ell^j)\right)\right)^{1-\ell_j}$$

The probability measure $\mathbb{P}_R$ on $(\Omega, 2^L)$ is then given by $\mathbb{P}_R(S) = \sum_{\ell \in S} p_R(S)$ for each $S \subseteq L$. We also define $\mathbb{E}_R$ to be the expectation under the probability measure $\mathbb{P}_R$.

At a high level, our proof relies on sample complexity lower bound for distinguishing between two distributions, formally defined as follows.

Definition 37 (Adaptive Classifier). Consider $R,B:[0,1]\to[0,1]$. Let (Ω,x) be a decision tree and $f:L(\Omega)\to\{R,B\}$. Then, (Ω,x,f) is called an *adaptive classifier* for R and B. Moreover, given constants $\alpha,\beta\in[0,1]$, an adaptive classifier (Ω,x,f) is called (α,β)-*confident* if

$$P_D := \mathbb{P}_R\left(f^{-1}(R)\right)\ge\alpha, \qquad\qquad (\textbf{D}\text{etection probability is high})$$

$$\text{and}\quad P_{FA} := \mathbb{P}_B\left(f^{-1}(R)\right)\le\beta. \qquad\qquad (\textbf{F}\text{alse-}\textbf{A}\text{larm probability is low})$$

Our lower bound results all rely upon a Theorem due to Wald and Wolfowitz (1948) for adaptive sequential hypothesis testing, which states that the *expected* number of samples collected in order to **adaptively** distinguish between a pair R,B (referred to as "red" and "blue") of distributions must be lower bounded by a function of α,β and the KL-divergence.

Theorem 28 **(Wald-Wolfowitz Theorem)**. *Consider $R,B:[0,1]\to[0,1]$ and an (α,β)-confident adaptive classifier (Ω,x,f). Denote $\Delta(R,B)=\max_{v\in\Omega}\mathrm{KL}\big(R(x_v),B(x_v)\big)$. Let $D(\ell)$ be the depth of leaf $\ell\in L(\Omega)$. Then,*

$$\mathbb{E}_R[D]\ge\frac{\alpha\log\frac{\alpha}{\beta}+(1-\alpha)\log\frac{1-\alpha}{1-\beta}}{\Delta(R,B)},\quad \textit{and}\quad \mathbb{E}_B[D]\ge\frac{\beta\log\frac{\beta}{\alpha}+(1-\beta)\log\frac{1-\beta}{1-\alpha}}{\Delta(B,R)}. \tag{43}$$

Subsequently, we apply this theorem to derive lower bounds for each of the following three regimes: $d=0$, $1\le d<\infty$ and $d=\infty$.

IV.5.2. Zero-Dimensional Family We first sketch the high level idea. Consider a policy π with $O(\log^2 T)$ regret. Fix a linear demand curve R, which we call the red curve, whose optimal price we denote p_R^*. For each t, we bound the expected regret in round t as follows. Choose $\Delta_t\sim\sqrt{\frac{\log T}{t}}$ and construct a blue linear demand function $B=B(t)$ whose optimal price is Δ_t greater than p_R^*. Consider the following classifier induced by the price choice of π: define the output of this classifier to be R if the price selected in round t is closer to p_R^*, and B otherwise.

Recall that the WW theorem asserts that if both the type I, II errors of a classifier are "low", then the expected numbers of samples under both hypotheses are "high". Now, on the one hand, since the policy has low regret, under B the probability for overshooting should be extremely low. On the other hand, we will consider consider small t, so that the number t of samples that the classifier collects is "low". Now, in order not to contradict WW Theorem, the error probability under R must be large! In other words, with considerable probability, the price selected at time t is greater than $p_R^*+\Delta_t/2$, hence a high regret is incurred under R in round t. Thus, by summing the expected regret from round $t=\log T$ to $\sqrt{T}$, we can lower bound the regret by

$$\sum_{t=1}^{\sqrt{T}}\Delta_t^2=\sum_{t=1}^{\sqrt{T}}\frac{\log T}{t}\sim\log^2 T.$$

We now present a formal proof. We will use the *contrapositive* version of Theorem 28.

Corollary 3. *Consider $R, B : [0,1] \to [0,1]$,*

$$\Delta(R,B) = \max_{v \in \Omega} \mathrm{KL}\left(\mathrm{Ber}\left(R(x_v), \mathrm{Ber}(B(x_v)\right)\right).$$

Suppose $0 \le \alpha < \frac{1}{2} < \beta \le 1$ and Let (Ω, x, f) be an (α', β')-confident adaptive classifier for R, B satisfying

(i) $\alpha' \le \alpha$, and

(ii)

$$\mathbb{E}_R[D] \le \frac{\alpha \log \frac{\alpha}{\beta} + (1-\alpha)\log\frac{1-\alpha}{1-\beta}}{\Delta(R,B)}, \quad \mathbb{E}_B[D] \le \frac{\beta \log \frac{\beta}{\alpha} + (1-\beta)\log\frac{1-\beta}{1-\alpha}}{\Delta(B,R)}.$$

Then, $1-\beta' \ge 1-\beta$.

Proof of Theorem 22. Let π be any markdown policy, which we also view as a decision tree in this proof. Consider a fixed reward function R with optimal price $p_R^* = \frac{1}{2}$. Fix $t \in [\log T, T^{1/2}]$, and let $\Delta_t = \sqrt{\frac{\log T}{t}}$. Consider a demand function $B = B_t$ given by $B_t(x) = 1 - (1+\Delta_t)x$. Note that the optimal price of B is $p_B^* = \frac{1}{2+\Delta_t}$, so $p_R^* + \Delta_t \le p_B^* \le p_R^* + 2\Delta_t$.

To apply Theorem 28, we consider the following classifier (Ω_t, x_t, f_t), induced by π. The node set Ω is simply all nodes in Ω of depth at most t, and x_t is simply the mapping $x : \Omega \to [0,1]$ restricted to Ω_t. Define $f_t : L(\Omega_t) \to \{R, B\}$ as follows:

$$f_t(\ell) = \begin{cases} B, & \text{if } x(\ell) > p_R^* + \frac{\Delta_t}{2}, \\ R, & \text{else.} \end{cases}$$

Let

$$\alpha = \mathbb{P}_B[f(\ell) = R], \quad \beta = \mathbb{P}_B[f(\ell) = B].$$

First we claim that if π has $O(\log^2 T)$ regret, then $\alpha \ge T^{-1/2}$. In fact, suppose the true revenue function is B and π selects a price lower than $p_B^* - \Delta_t/2$. Then, due to the markdown constraint, an $\Omega(\Delta_t^2)$ regret is incurred in each future round and hence the regret is $\Omega(\alpha \Delta_t^2 T)$. Thus, to achieve $O(\log^2 T)$ regret, we need $\alpha \Delta_t^2 T \le \log^2 T$, i.e.

$$\alpha \le \frac{t \log T}{T} = \tilde{O}(T^{-1/2}).$$

We next conclude the proof using Corollary 3. Recall that $\log T \le t \le \sqrt{T}$, so

$$t < \frac{\log T}{\Delta_t^2} \le \frac{3 \log T}{\Delta(R,B)},$$

where the last step follows since $\max_x \mathrm{KL}(R(x), B(x)) \le \Delta_t^2/3$. By Corollary 3, since $\alpha = T^{-1/2}$, and $t = \mathbb{E}_R[D] < \frac{\log T}{\Delta(R,B)}$, we have $1 - \beta \ge \frac{1}{4}$. In other words, when R is the true revenue function, w.p. $\frac{1}{4}$ the price at t will be higher than $p_R^* + \Delta_t$, hence the expected regret at this round is $\Omega(\Delta_t^2) = \Omega(\frac{\log T}{t})$.

Finally, since the above argument holds for all $t \in [\log T, T^{1/2}]$, the total regret under R is lower bounded as

$$\begin{aligned}
\mathrm{Reg}(\pi, R) &\geq \sum_{t=\log T}^{\sqrt{T}} \frac{\log T}{t} \\
&= \log T \cdot \left(\sum_{t=1}^{\sqrt{T}} \frac{1}{t} - \sum_{t=1}^{\log T} \frac{1}{t} \right) \\
&= \Omega(\log^2 T) - O(\log T \cdot \log\log T) = \Omega(\log^2 T),
\end{aligned}$$

and the proof is complete. □

IV.5.3. Finite-Dimensional Family We now prove Theorem 24. We first describe our proof at a high level. For each d we construct a pair of demand functions D_{blue}, D_{red} on price space $[\frac{1}{2}, 1]$ with $D_{blue}(1) = D_{red}(1)$. Moreover, price 1 is the unique optimal price of D_{blue} and suboptimal for D_{red}. Since the gap between these two demand functions is tiny near price 1, to distinguish between them we have to reduce the price away from 1. Thus learner faces the following trade-off: if she reduces the price by too much, then a high regret is incurred under D_{blue} since its optimal price is at 1; otherwise, the difference between these two curves is too small and she has to explore for too many rounds near price 1, which is suboptimal for D_{red}, hence incurring a high regret.

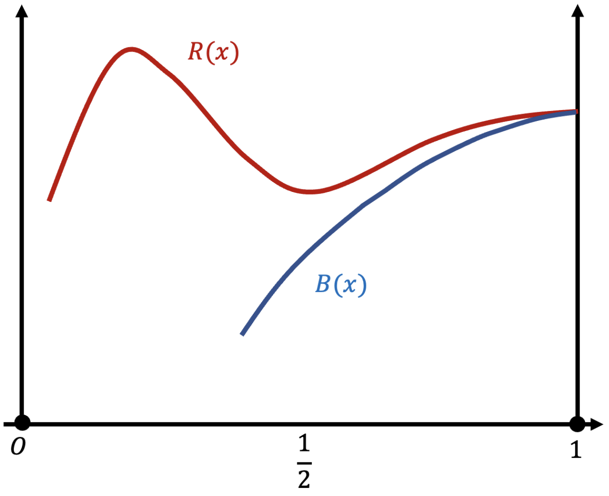

Figure 8 **Illustration of Lemma 36.**

Consider a policy with low regret. Choosing suitable neighborhood $[1-h,1]$, we convert this policy into a classifier that returns R_r or R_b based on whether the price in round $\frac{T}{4}$ is within this neighborhood. We first argue that if the policy has low regret, then it has to perform reasonably well on this classification problem, otherwise an $\Omega(h^2 T)$ regret is incurred. Then, we use the generalized Wald-Wolfowitz Theorem (Theorem 28) to show that in order to distinguish between these two curves, the policy has to spend $\Omega(h^{-2d})$ rounds inside the neighborhood in expectation, incurring $\Omega(h^{-2d}T)$ regret under R_{red}.

Now we formalize the above ideas. As the key step, we first explicitly construct a pair of demand curves with the following properties, and then show that distinguishing between this pair of reward functions requires many samples, hence incurring a high regret.

Lemma 36. *For any $d \geq 1$, there exists a pair D_{red}, D_{blue} of degree-d polynomial demand functions satisfying the following properties.*

1. **Monotonicity:** *Both are non-increasing on* $[1/2,1]$*,*
2. **First Order Optimality:** *Denote $R_i(x) = x \cdot D_i(x)$ for $i \in \{red, blue\}$, then $\max_{x\in[1/2,1]} R_{\text{blue}}(x)$ is attained at $x=1$. Moreover, $R'_{\text{blue}}(1)=0$,*
3. **Interior Optimal Price:** *The function R_{red} is maximized at some price $x \in [0, \frac{1}{2}]$,*
4. **Hardness Of Testing:** *Let $Gap(h) = \max_{x\in[1-h,1]}\{|D_{red}(x) - D_{blue}(x)|\}$, then $Gap(h) \leq O(h^d)$ as $h \to 0^+$. In particular, this implies that $R_{\text{blue}}(1) = R_{\text{red}}(1)$.*

Proof. The proof involves explicit construction of the desired families of demand functions. In the next two subsections, we consider the case $d=1$ and $d \geq 2$ separately.

Step 1. Suppose $d=1$. Let $p_{min} = 1/2, p_{max} = 1$. Consider the demand functions

$$D_{\text{blue}}(1-h) = 1+h, \quad D_{\text{red}}(1-h) = 1+5h,$$

or, substituting $x = 1-h$,

$$D_{\text{blue}}(x) = 2-x, \quad D_{\text{red}}(x) = 6-5x.$$

Let us verify each of the four conditions in Lemma 36:

1. both curves are clearly strictly decreasing.
2. $R_0(x) = x(2-x)$, so $R'_0(x) = 2-2x$. So its unique local maximum is attained at $x=1$. Moreover, $R''_0(1) = -2 < 0$.
3. $R'_1(x) = 6-10x$, so R_1 attains maximum at $x = 3/5$.
4. $|D_{\text{blue}}(1-h) - D_{\text{red}}(1-h)| = 4-4h = O(h)$.

Step 2. Suppose $d \geq 2$. In this case, consider the following two demand functions:

$$D_b(M-h) = 1 + h + bh^d, \quad D_r(M-h) = 1 + h + rh^d,$$

defined on the interval $[0, M]$ where M will be chosen to be some large number soon. The proof then follows by replacing h with Mh, hence re-scaling the domain to $[0, 1]$.

We first verify some trivial properties. Note that $D_b(1-h) - D_r(1-h) = (r-b)h^d$, so the gap between these two demand functions around price M is on the order of $O(h^d)$, and hence the last condition is satisfied.

We next verify that when $b = M^{-d}$, the function $R_b(x)$ attains maximum at $x = M$, formally, for any $d \geq 2$, it holds $\bar{R}_b(h) \leq M$ for any $h \in [0, M]$. To show this, observe that

$$\begin{aligned}\bar{R}_b(h) \leq M &\iff M - \frac{1}{M}h^2 + \left(\frac{h}{M}\right)^d (M-h) \leq M \\ &\iff \left(\frac{h}{M}\right)^d (M-h) \leq \frac{1}{M}h^2 \\ &\iff \left(\frac{h}{M}\right)^{d-1} (M-h) < h.\end{aligned}$$

To show the above holds for all $h \in [0, M]$, we rescale h by setting $h = \rho M$, where $\rho \in [0, 1]$. Then, the above becomes

$$\left(\frac{\rho M}{M}\right)^{d-1} (M - \rho M) < \rho M,$$

i.e.

$$\rho^{d-2}(1-\rho) < 1,$$

where clearly holds for all $\rho \in [0, 1]$ when $d \geq 2$.

We finally verify that maximum of $R_r(x)$ is attained in the interior of $[0, M]$. First note that $R_r(0) = 0$ and $R_r(M) = 1$, so it suffices to show that $\max_{x \in [0,M]} R_r(x) > 1$. To this aim, note that $R_r(M-h) - R_b(M-h) = (r-b)h^d$, and the proof follows. □

It will be convenient for the proof to only consider policies represented by trees where the node prices never change after $\frac{T}{2}$.

LEMMA 37 **(Jia et al. (2021).)**. *For any markdown policy $\mathbb{A}$, there is a policy $\mathbb{B}$ which makes no price change after $\frac{T}{2}$ such that* $\mathrm{Reg}(\mathbb{B}, R) \leq 2 \cdot \mathrm{Reg}(\mathbb{A}, R)$ *for all $R \in \hat{\mathcal{F}}_L$.*

By Lemma 25, we may consider only policies which makes no price changes after $\frac{T}{2}$. we first construct a classifier (Ω', x', f') as follows. With some foresight choose $h = T^{-\frac{1}{2d+2}}$. Let $\Omega' = \{v \in \Omega : d(v) \leq \frac{T}{4},$ and $x(v) \leq 1 - h\}$, and $x' = x|_{\Omega'}$. Here, the reason for choosing $\frac{T}{4}$ in the step above is we need to leave enough rounds for exploitation after going below price $1-h$. Define $f:\Omega' \to \{red, blue\}$ as

$$f(\ell) = \begin{cases} B, & \text{if } x(\ell) > 1-h, \\ R, & \text{else.} \end{cases}$$

Recall that $\mathbb{P}_i$ denotes the distribution over the leaves of $\mathbb{T}$ under color $i \in \{red, blue\}$, and that $(\mathbb{T}, f)$ is (α,β)-confident if

$$\mathbb{P}_{blue}(f(\ell) = red) \le \alpha, \text{ and } \mathbb{P}_{red}(f(\ell) = red) \ge \beta.$$

We first show that if $\mathbb{A}$ has the target regret, then $\mathbb{T} := (\Omega', x', f';)$ has to be $(1/3, 2/3)$-confident. Formally, we have the following lemma. Recall that $N(a,b)$ is the number of rounds the policy stays in price interval $[a,b]$.

Lemma 38. *If* $\mathrm{Reg}(\mathbb{A}) \le T^{\frac{d}{d+1}}$*, then* $\mathbb{T}$ *is* $(\frac{1}{3}, \frac{2}{3})$*-confident.*

Proof. There are two cases. Suppose $\mathbb{P}_R[f(L) = B] \ge \frac{1}{3}$, then the policy selects prices in $[1-h, h]$ for $\frac{T}{4}$ rounds. Thus, $\mathbb{E}_R[N(1-h,h)] \ge \frac{T}{4} \cdot \frac{1}{3} = \frac{T}{12}$. In the other case, suppose $\mathbb{P}_B[f(L) = R] \ge \frac{1}{3}$, then with probability $\frac{1}{3}$ the price becomes lower than $1-h$. Note that $R'_B(1) = 0$, so by Taylor expansion, an $\Omega(h^2)$ regret is incurred in each future round. Since there are $\frac{T}{4}$ rounds remaining, the total regret in this case is at least $\Omega(h^2 T)$. □

Then, we show that if $\mathbb{T}$ decides the color correctly with high confidence, then we must spend many rounds (in expectation) in $[1-h, 1]$. In fact, by Theorem 28 and noting that $\mathrm{KL}(R_{\mathrm{red}}(x), R_{\mathrm{blue}}(x)) \le h^{2d}$, we immediately obtain the following.

Lemma 39. *Let* $D(\ell)$ *be the level of* $\ell \in L(\mathbb{T})$*. Suppose* $\mathbb{T}$ *is* $(\frac{1}{3}, \frac{2}{3})$*-confident, then*

$$\mathbb{E}_R[D] = \Omega(h^{-2d}).$$

Note that the regret per round in $[1-h, 1]$ under D_{red} is $\Omega(1)$, thus for any algorithm π with $O(T^{\frac{d}{d+1}})$ regret, by Lemma 38 and 39,

$$\mathrm{Reg}(\pi, R) \ge \mathbb{E}_R[N(1-h,h)] \cdot \Omega(1) \ge h^{-2d} \cdot \Omega(1) = \Omega(T^{\frac{d}{d+1}}),$$

and Theorem 24 follows. □

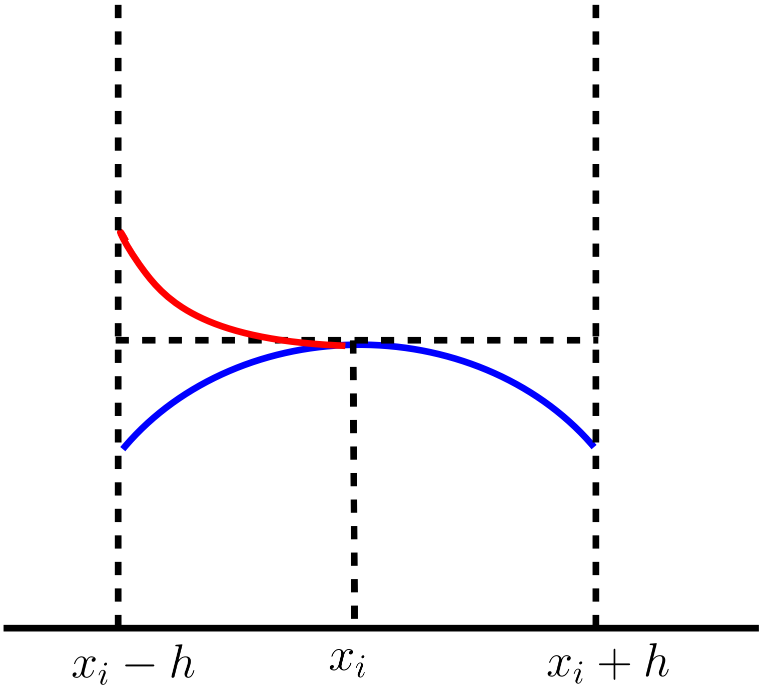

Figure 9 **Bow-shaped (blue) and S-shaped (red) reward curves**

IV.5.4. Infinite-Dimensional Family We next show Theorem 26. The proof uses similar idea as in the lower bound proof in Jia et al. (2021). However, for each $s \geq 2$ we need to construct an s-sensitive family of demand functions.

We consider the following s-sensitive family of unimodal reward curves. With some foresight, choose $h = T^{-\frac{1}{3s+1}}$. In the construction, we will use the following S-shaped curves (or S *curves*) and bow-shaped curves (or B *curves*), as shown in Figure 9.

We now formally describe those S and B curves. For simplicity we assume $m := \frac{1}{h}$ is an even integral. Define a decreasing sequence $x_i = 1 - (2i-1)h$ for each $i = 1, ..., m/2$ of prices. Each pair of curves B_i, S_i are defined in the interval $[x_i - h, x_i + h]$. These two curves are identical at higher prices than x_i and, scanning from right to left, start to diverge at a rate of h^s starting at x_i. Formally,

$$B_i(x_i + \xi) = y_i - |\xi|^s, \quad \forall \xi \in [-h, h],$$

and

$$S_i(x_i + \xi) = \begin{cases} y_i + |\xi|^s, & \text{if } \xi \in [-h, 0], \\ y_i - |\xi|^s, & \text{if } \xi \in [0, h], \end{cases}$$

where $y_i = \frac{1}{2} + 2ih^s$.

Now we are ready to construct the reward functions using these gadgets. For $i = 1, ..., m/2$, scanning from prices high to low, the reward function R_i is a concatenation of $(i-1)$ consecutive

S-curves, followed by one B curve, and finally a curve extending downwards the left portion of B until reaching the x-axis. Formally for any $i = 1, ..., \frac{m}{2}$,

$$R_i(x) = \begin{cases} S_j(x), & \text{if } x \in [x_j - h, x_j + h] \text{ for } j \leq i - 1, \\ B_i(x), & \text{if } x \in [x_i - h, x_i + h], \\ sh^{s-1}x + (y_i - h^s - sh^{s-1}(x_i - h)) & \text{if } x \leq x_i - h. \end{cases}$$

Finally, we need a special reward function R_0, that consists only of S-curves on $[\frac{1}{2}, 1]$, and extends upwards when the prices moves below $\frac{1}{2}$, analogous to the construction to the roof curves in the lower bound proof of Theorem 13 in the previous chapter. Formally,

$$R_0(x) = \begin{cases} R_{\frac{m}{2}}(x), & \text{if } x \geq 1/2, \\ y_{\frac{m}{2}} + (x_{\frac{m}{2}} - x)^s, & \text{if } x \in [0, 1/2]. \end{cases}$$

The lower bound is again showed using the Wald-Wolfowitz Theorem (Theorem 28). At a high level, any reasonable policy π needs to solve a hypothesis testing problem in each interval $[x_i - h, x_i + h]$, which aims at distinguishing between R_0 and R_i. Note that R_0 and R_i are completely identical on prices higher than x_i, and only starts to differ on prices lower than x_i, at a rate of h^s. Hence, the maximum KL divergence on this interval is $\sim h^{2s}$, and by the Wald-Wolfowitz Theorem (Theorem 28), in expectation $\Omega(h^{-2s})$ samples are necessary assuming the policy π is able to distinguish between these two reward functions.

To see why π *must* be able to distinguish between the two curves, for the sake of contradiction, suppose otherwise, say, under true reward curve R_i the policy π has a high probability of mistakenly return R_0 as the true curve, and hence reduces the price below $x_i - h$, incurring an $\Omega(h^s)$ regret per round. This leads to an $\Omega(h^s T) = \Omega(T^{\frac{2s+1}{3s+1}})$ regret in future rounds, contradicting the low-regret assumption (with suitably chosen constants). Since the number of intervals is $\Omega(h^{-1})$, we have

$$\mathrm{Reg}(\pi, R_0) \geq \Omega(h^{-2s}) \cdot \Omega(h^{-1}) = \Omega(h^{-1-2s}) = T^{\frac{2s+1}{3s+1}},$$

and the proof follows. $\square$

Chapter V Short-Lived High-Volume Bandits: Algorithms and Field Experiment

Consider the problem of recommending newly-created content. For *long-lived* content, the problem is arguably straightforward: spend a negligible amount of time collecting sufficient data in the form of user feedback, and then apply a suitable offline predictive model. For *low volume* content relative to the number of recommendation, the problem is also well-understood: dedicated exploration methods (e.g. A/B testing) are sufficient for determining which content to show. The question then is, how should an online platform decide what content to display to each user? In addition to the well-known "learning-vs-earning" trade-off in multi-armed bandit (MAB) models, the online platform needs to resolve an additional concern: the balance between the exploration of newly arriving and older contents. We propose a simple bandit-based approach that not only settles this challenge but can be easily implemented in practice. We implemented this policy in a live field experiment with a large lockscreen content platform which faces exactly this challenge. Over the course of a field experiment running over two weeks, our policy achieved an $6 \sim 12\%$ improvement in conversion rates, relative to a neural network based control policy.

V.1. Introduction

There has been a long history where online platforms leverage the scale of data, especially user attention, to make better decisions for newly-arriving products or contents. By and large, recommendation tasks can be classified into four categories based on the *lifetime* and *volume* of contents generated (see Figure 10). For persistent (long-lived) content, the problem is arguably straightforward: spend a small amount of time collecting sufficient data in the form of user feedback, and then a suitable offline predictive model, which might range in sophistication from a basic collaborative filtering algorithms to, nowadays, deep neural networks (DNNs). For example, recently YouTube deployed a recommender system comprised of two deep neural networks: one for candidate generation and one for ranking (Covington et al. (2016)).

Orthogonal to content lifetime, when there is a *low volume* of content relative to the number of users, the problem is similarly well-understood: dedicated exploration methods (e.g. A/B testing) are sufficient for finding the right segments of users for which the content is most appealing. LinkedIn runs over 400 concurrent experiments per day to compare different designs of their website, with the goal of, for example, encouraging users to better establish their personal profile, or increasing the subscriptions to LinkedIn Premium (Xu et al. (2015)).

Naturally then, the most challenging settings are where the content to be recommended is *short-lived* and *high-volume*. Such settings arise, for example, in content aggregation platforms (e.g.

Apple News) and platforms with content that is entirely user-generated (e.g. TikTok). In these settings, both of the previous approaches are prone to failure: offline predictive algorithms do not receive enough data on individual content to achieve meaningful accuracy due to the short lifetime, and dedicated exploration methods are ill-suited to the high volume of contents.

In practice, platform such as Tiktok, Google and Kwai have deployed DNN-based recommender system for short-lived contents, and frequently re-train the DNN by incorporating the latest data. However, both retrieval of data and retraining of NN requires considerable amount of time and space. This poses substantial challenge for the companies in terms of both human and computational resources. To minimize the resources used for exploration, it is better to instead focus on recommendation policies that are operationally simple and statistically interpretable.

We investigate this problem through collaboration with Glance, a large content-aggregation platform who is faced with exactly this challenge. More precisely, Glance produces a high volume of "Glance cards" (see Figure 10), which consists of a background picture carefully crafted by their marketing team, and a link pointing to an external information source.

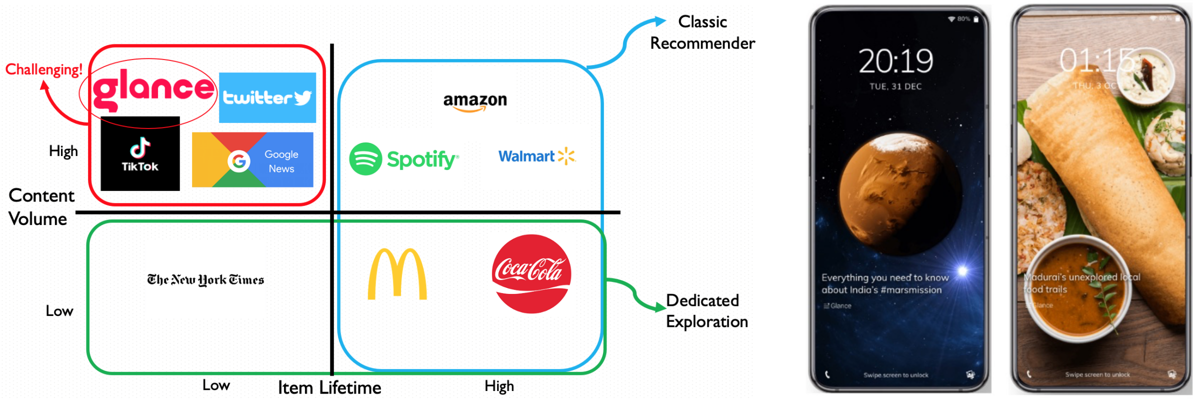

Figure 10 **Left: Recommendation tasks may be categorized by lifetime and volume. Right: An example of lockscreen cards created by Glance.**

Multi-Armed Bandits (MAB, or simply "bandits") provide a good framework for such policies. On the one hand, compared to DNN's, bandit policies are much more interpretable. On the other hand, bandit policies usually involves simple computation or sampling, and as a result, they are easier to code, maintain, and above all, computationally fast. We formulate the problem as an MAB model where in each round, a set of arms, which model the newly-generated contents, arrives with unknown mean conversion rates, and are available for a given short period of time. The platform then selects and serves a set of arms to each user, and observes the conversion rates of the selected arms.

As opposed to most previous work on MAB problems where the worst case input is considered, in this work we assume that the reward rates are independently and identically distributed (i.i.d.) with a known distribution. The reason is two-fold. First, this assumption better captures the uncertainty in conversion rates in reality compared to the adversarial model. Further, it brings extra *structure* that the learner may utilize for balancing the exploration between arms of different ages. In this work, we will for simplicity consider D being uniform distribution, but all of the four results in Table 10 can be generalized to distributions D such that D (i) has a finite support $[a, b]$, and (ii) admits a cumulative density function (cdf) F with $1 - F(b - \varepsilon) = O(\varepsilon)$ for any $\varepsilon \in [0, 1]$.

V.1.1. Our Contributions Our first contribution is formulating a problem that models the recommendation problem for short-lived high volume items, faced by many online platforms nowadays. To be more precise, we consider a batched bandits model where arms are arriving and expiring over time, with unknown reward rates that are drawn from the uniform distribution. We show two lower bounds. Suppose there are K new items arriving each period, and n *identical*, static users on the platform. We first present a general $\Omega\left(\frac{1}{K}\right)$ lower bound that holds for any n, K. However this lower bound becomes very weak when K is large compared to n. This motivates us to consider the case where $K = \Omega(\sqrt{n})$, where we show an $\Omega(n^{-1/2})$ lower bound.

Our second contribution is proposing a novel policy called the *Sieve Policy.* The policy iteratively removes the arms that are unlikely to be optimal in its cohort, based on the current reward estimate, and hence focuses on finer estimates of the remaining arms. We prove that an ℓ-layered Sieve Policy admits regret $O\big(\frac{1}{K} + (\frac{K}{n})^{\frac{\ell}{\ell+2}}\big)$, which decreases as we increase the number ℓ of layers. Furthermore, when $K = \Omega(\sqrt{n})$, by randomly sampling the arms, the regret of our sieve policy becomes $O(n^{-\frac{\ell}{2(\ell+1)}})$, which approaches the aforementioned lower bound $\Omega(n^{-1/2})$ as ℓ increases.

As our third contribution, we collaborated with Glance, an Indian lock-screen content platform who faces exactly this challenge, to implement a basic version of our Sieve Policy in a large-scale field experiment. Glance generates hundreds of *content cards* on an hourly basis, most of which are available for at most 24 hours. They deployed a state-of-the-art Deep Neural Network (DNN) based recommender system, which is time-consuming to re-train and hence unable to utilize user feedback in a timely manner. In a live field experiment, we observed that our 1-layer sieve policy, with minor adaptations for practical concerns, outperforms their DNN-based recommender system by 6% in the number of impressions per user and 12% in the number of conversions per user.

	$K < \sqrt{n}$	$K \geq \sqrt{n}$
Lower Bound	$\frac{1}{K}$	$\frac{1}{\sqrt{n}}$
Upper Bound	$\left(\frac{K}{n}\right)^{\frac{\ell}{2\ell+2}}, \quad \forall \ell \leq W$	$\left(\frac{1}{\sqrt{n}}\right)^{-\frac{W}{2W+2}}$

Table 10 Our Results

V.1.2. Related Literature A central problem in online platforms is content recommendation. Various techniques from different fields have been applied into designing better recommender systems, including Deep Neural Networks (DNN) Aggarwal et al. (2016), and collaborative filtering Koren and Bell (2015). Our key challenge is recommending short-lived contents, especially in the face of high volume of contents. Since many firms are using Deep Neural Network (DNN) based recommenders, a natural approach for recommending short-lived contents would be simply be updating the DNN more frequently. While this approach has been adopted by leading platforms such as Tiktok and Youtube, it is less realistic for relatively smaller platforms. In fact, frequent updates of neural network involves retrieving the latest interaction data and retaining the DNN, both of which can be substantially time consuming, and may require considerable expertise.

In contrast, online learning, or more specifically, the Multi-Armed Bandits (MAB) model, provides an alternate framework for designing recommenders based on its simplicity and interpretability. In the MAB problem (e.g. Lai and Robbins (1985)), the learner is given a set of *arms*, each associated with an unknown distribution, from which a *reward* is independently drawn each time the arm is selected. The goal is to sequentially select arms so as to maximize the total reward.

There is a growing literature on bandit-based policy in operations management and marketing. For instance, Li et al. (2010) formulated the news article recommendation problem as a contextual bandits problem, where they represented the users and news article as *feature vectors*, and the expected reward when an article is recommended to a user is assumed to be a logistic function of the inner product between the feature vectors. Bouneffouf and Rish (2019) summarized some other practical applications of bandit-based algorithms, from recommender systems and information retrieval to healthcare and finance.

There are two MAB variants most closely related to our problem. The first is *mortal bandits* (Chakrabarti et al. (2008), Levine et al. (2017)), where each arm has a stochastic *lifetime* after which it becomes unavailable. The other variant is *Batched Bandits* (Perchet et al. (2016), Gao et al. (2019), Agarwal et al. (2017)). In this variant, the learner needs to select arms in batches, subject to a given budget on the total number of batches she could use, and another budget on the total number arms to be selected. In particular, the learner is allowed to determine the size of each batch, rendering the technique inapplicable to the analysis of our problem. In fact, in practice the batch sizes correspond to the number of interactions observed per period, which the decision-maker has no control over in our setting.

Recently there is a growing literature on evaluating bandit-based policies via field experiments on real systems. For example, Schwartz et al. (2017) considered how to allocate percentages of impressions to each new ad, so as to maximize customer acquisition. They implemented a Thompson sampling based policy in a live field experiment with a large retail bank, and observed a significant increase in the customer acquisition rate. Ye et al. (2020) modeled the cold-start problem for online advertising as contextual bandit, where the platform needs to trade off between the short-term revenues and long-term market thickness. They demonstrated the efficacy of their policies via a field experiment on a leading video-sharing platform.

V.1.3. Organization In Section V.2, we formally formulate the problem as a variant of MAB, and introduce a lower bound on the regret. Then in Section V.3, we present our Sieve Policy and upper bound its regret. Finally, we provide the full details and results of our field experiments in Section V.4.

V.2. Model and Lower Bound

V.2.1. Formulation Consider a *Multi-Armed Bandits* (MAB) formulation of our problem. At the start of each round $t = 1, 2, ...$, a set A_t of K arms arrives, each arm a with an unknown *reward rate* $\mu(a)$. To model the transient nature of the recommended content, each arm is associated with a known *lifetime* $W > 0$, representing the number of rounds the arm remains *available.* Thus, in round t, the set of available arms are $\cup_{\tau=t-W}^{t} A_\tau$. For notational convenience, for $0 < s < t$ we denote $A_s^t = \cup_{\tau=s}^{t} A_\tau$, and define $A_s = \emptyset$ if $s \leq 0$.

The learner in each round t selects a multi-set of n available arms in a *batched* manner. The random *reward* of each selected arm is independently drawn from $\text{Ber}\left(\mu(a)\right)$, and observed by the learner. For concreteness, one may assume that there are n *identical* users in the platform, each to be assigned to play exactly one arm in each round, and generates a random reward. A *policy* for decision making can be formally specified by a sequence $\pi = \{\pi_t : t = 1, 2, ...\}$ of mappings where $\pi_t(a)$ denotes the number of times arm a is selected at time t.

We measure the loss of a policy by comparing it against the optimal policy that knows $\mu(a)$ beforehand, in which case the policy simply selects the arm with the maximum reward rate for n times, collecting an expected reward of $n \cdot \mu_{max}(A_{t-W}^t)$, where we denote $\mu_{max}(A) = \max_{a \in A} \mu(A)$ for any set A of arms.

As opposed to most work on MAB where the benchmark is the worst-case input, we study a more realistic scenario where the reward rates are assumed to be i.i.d drawn from a known, fixed distribution D, and consider the performance of a policy on an "average" instance. More precisely, we will consider an objective called the *long-term average regret*. Formally, for policy π define

$$\mathrm{Reg}(\pi, T) = \frac{1}{nT}\mathbb{E}\left[\sum_{t=1}^{T} \sum_{a\in A_{t-W}^{t}} \pi_t(a)\cdot\left(\mu_{max}\left(A_{t-W}^{t}\right) - \mu(a)\right)\right],$$

where the expectation is over the input as well as the random rewards. To measure the regret in the long-run, we take limit and define the long-term average regret as

$$\mathrm{Reg}(\pi) = \overline{\lim_{T\to\infty}}\,\mathrm{Reg}(\pi, T).$$

Compared to the worst-case analysis, this average case analysis not only enables us to achieve richer theoretical results, but more importantly, better captures the reality and provides more insights towards how to explore the arms in the face of short lifetime. In this work, we assume D to be the uniform distribution on $[0,1]$, though it is straightforward to extend our results to more general distributions.

V.2.2. Lower Bounds: High Level Ideas We start with a simple $\Omega\left(\frac{1}{K}\right)$ lower bound. Recall that the reward rates of the arms are drawn from the uniform distribution. At each round t, there is a $\frac{1}{W}$ probability that the reward-maximizing available arm a_t^* is contained in the arriving batch A_t of arms. The policy now faces the following dilemma. Suppose the policy selects arms from A_t for many times. Since it has no knowledge about A_t except that the reward rates are uniform, the policy can not do much better than randomly guessing, incurring a high regret in this round.

On the other side, suppose the policy selects arms from A_t for very few times. Recall that A_t contains a_t^* with probability $\frac{1}{W}$. Moreover, due to the uniform reward rate assumption, when the above event occurs one should naturally expect $\mu(a^*)$ to be $\sim \frac{1}{KW}$ higher than the optimal arm in A_{t-1}^{t-W}. In this case, the regret incurred for selecting an arm from A_{t-1}^{t-W} is $\sim \frac{1}{WK}\cdot\frac{1}{K} = \frac{1}{W^2K}$, and we obtain our first lower bound, as formally stated below.

Theorem 29. *For any policy π, we have* $\mathrm{Reg}(\pi) \ge \frac{1}{12W^2K}$.

However, this bound becomes weaker when K is large, so we next focus on the case when K is "large" compared to n. Somewhat interestingly, we establish a lower bound which transitions from the above $\frac{1}{K}$ lower bound continuously to the large K regime. At a high level, we argue that if a policy has regret, then it has to "identify" a nearly-optimal arm, and hence has to explore many distinct arms, wherein a high regret is incurred. We formally state the second lower bound below.

Theorem 30. *Suppose $n \le K^2$, then for any policy π we have* $\mathrm{Reg}(\pi) \ge \frac{1}{12W\sqrt{n}}$.

V.2.3. The First Lower Bound. We first describe the high level idea. Suppose

We now make the above ideas precise. Consider the event

$$G_{t-1} = \left\{1 - \frac{2}{(W-1)K} \leq \mu_{max}(A_{t-W}^{t-1}) \leq 1 - \frac{1}{(W-1)K}\right\}$$

that the input instance is "well-behaved". We first show that this event occurs with large probability.

LEMMA 40. *If $K \geq 10$ then $\mathbb{P}(G_{t-1}) \geq \frac{1}{8}$.*

Proof. Denote $\mu_{\max} = \mu_{max}(A_{t-W}^{t-1})$. Let

$$H = \left\{\mu_{\max} \geq 1 - \frac{1}{(W-1)K}\right\} \text{ and } H' = \left\{\mu_{\max} < 1 - \frac{2}{(W-1)K}\right\}.$$

Then for $K \geq 10$, we have

$$\mathbb{P}\left[\overline{H}\right] = \mathbb{P}\left[\mu_{\max} < 1 - \frac{1}{(W-1)K}\right] = \left(1 - \frac{1}{(W-1)K}\right)^{(W-1)K} \geq \frac{3}{4} \cdot e^{-1},$$

and thus $\mathbb{P}[H] \leq 1 - \frac{3}{4e}$. Moreover,

$$\mathbb{P}[H'] = \left(1 - \frac{2}{(W-1)K}\right)^{\frac{(W-1)K}{2} \cdot 2} \leq e^{-2}.$$

Therefore,

$$\mathbb{P}[G_{t-1}] \geq 1 - \mathbb{P}[H'] - \mathbb{P}[H] \geq 1 - \left(1 - \frac{3}{4e}\right) - e^{-2} > \frac{1}{8},$$

and the proof follows. □

We next lower bound the regret at time t conditional on G_{t-1}. For each t and event A, we write $\mathbb{P}_t(A) = \mathbb{P}[A|G_{t-1}]$ and $\mathbb{E}_t(A) = \mathbb{E}[A|G_{t-1}]$. To analyze the above terms, we introduce the following events. Consider the event

$$\mathcal{E}_t = \left\{\sum_{a \in A_t} \pi_t(a) \geq \frac{n}{2}\right\}$$

be that π selects arms from A_t at time t for at least $\frac{n}{2}$ times. Further, define

$$E_t^- = \left\{\mu_{max}(A_t) \leq \mu_{max}(A_{t-W}^{t-1}) - \frac{1}{K}\right\} \text{ and } E_t^+ = \left\{\mu_{max}(A_t) \geq \mu_{max}(A_{t-W}^{t-1}) + \frac{1}{WK}\right\}.$$

One can verify that both events are likely to occur conditional on G_{t-1}, as formally stated below.

LEMMA 41. *$\mathbb{P}_t[E_t^-] \geq \frac{1}{2}$ and $\mathbb{P}_t[E_t^+] \geq \frac{1}{W}$.*

Intuitively, the regret is high at time t under the following two circumstances: (i) the policy selects many arms from A_t but the optimal arm is not in A_t, and (ii) the policy selects A_t very few times but the optimal arm is from A_t. We next characterize the regret under these two scenarios.

LEMMA 42. *It holds that* $\mathbb{E}_t[R_t|\mathcal{E}_t \cap E_t^-] \geq \frac{n}{2K}$ *and* $\mathbb{E}_t(R_t|\overline{\mathcal{E}}_t \cap E_t^+) \geq \frac{n}{2WK}$.

Proof. Write $\mu_t^* = \mu_{\max}(A_t)$. Consider the first inequality. Conditional on E^-, the optimal arm is not in A_t, and since $\mathcal{E}_t$ occurs, i.e. the policy selects arms from A_t for $\frac{n}{2}$ times, a high regret is incurred. We formalize this idea as follows.

$$\begin{aligned}
\mathbb{E}_t[R_t|\mathcal{E}_t \cap E^-] &= \sum_{a \in A_{t-W}^t} \mathbb{E}_t[\pi_t(a) \cdot (\mu_t^* - \mu(a))|\mathcal{E}_t \cap E^-] \\
&\geq \sum_{a \in A_t} \mathbb{E}_t[\pi_t(a) \cdot (\mu_t^* - \mu(a))|\mathcal{E}_t \cap E^-] \\
&\geq \sum_{a \in A_t} \mathbb{E}_t[\pi_t(a)|\mathcal{E}_t \cap E^-] \cdot \mathbb{E}_t[\mu_t^* - \mu(a)|\mathcal{E}_t \cap E^-] \\
&\geq \left(\sum_{a \in A_t} \mathbb{E}_t[\pi_t(a)|\mathcal{E}_t \cap E^-] \right) \cdot \mathbb{E}_t\left[\mu_t^* - \mu_{\max}(A_t)|\mathcal{E}_t \cap E^-\right] \\
&\geq \frac{n}{2} \cdot \frac{1}{K} = \frac{n}{2K}.
\end{aligned}$$

Now we show the second inequality. Conditional on E^+, the optimal arm is in A_t, and since $\overline{\mathcal{E}}_t$ also occurs, i.e. the policy selects arms in A_{t-W}^{t-1} for more than $\frac{n}{2}$ times, a high regret is incurred. We formalize this idea below.

$$\begin{aligned}
\mathbb{E}_t[R_t|\overline{\mathcal{E}_t} \cap E^+] &= \sum_{a \in A_{t-W}^t} \mathbb{E}_t[\pi_t(a) \cdot (\mu_t^* - \mu(a))|\overline{\mathcal{E}_t} \cap E^+] \\
&\geq \sum_{a \in A_{t-W}^{t-1}} \mathbb{E}_t[\pi_t(a) \cdot (\mu_t^* - \mu(a))|\overline{\mathcal{E}_t} \cap E^+] \\
&\geq \sum_{a \in A_{t-W}^{t-1}} \mathbb{E}_t[\pi_t(a)|\overline{\mathcal{E}_t} \cap E^+] \cdot \mathbb{E}_t[(\mu_t^* - \mu(a))|\overline{\mathcal{E}_t} \cap E^+] \\
&\geq \left(\sum_{a \in A_{t-W}^{t-1}} \mathbb{E}_t[\pi_t(a)|\overline{\mathcal{E}_t} \cap E^+] \right) \cdot \mathbb{E}_t\left[(\mu_t^* - \mu_{\max}(A_{t-W}^{t-1}))|\overline{\mathcal{E}_t} \cap E^+\right] \\
&\geq \frac{n}{2} \cdot \frac{1}{WK} = \frac{n}{2WK},
\end{aligned}$$

and the proof follows. □

Proof of Theorem 29. Observe that

$$\begin{aligned}\mathbb{E}_t(R_t) &= \mathbb{E}_t(R_t|\mathcal{E}_t \cap E^-)\cdot\mathbb{P}_t(\mathcal{E}_t\cap E^-) + \mathbb{E}_t(R_t|\mathcal{E}_t\cap E^+)\cdot\mathbb{P}_t(\mathcal{E}_t\cap E^+)\\ &\quad + \mathbb{E}_t(R_t|\overline{\mathcal{E}}_t\cap E^-)\cdot\mathbb{P}_t(\overline{\mathcal{E}}_t\cap E^-) + \mathbb{E}_t(R_t|\overline{\mathcal{E}}_t\cap E^+)\cdot\mathbb{P}_t(\overline{\mathcal{E}}_t\cap E^+)\\ &\ge \mathbb{E}_t[R_t|\mathcal{E}_t\cap E^-]\cdot\mathbb{P}_t(\mathcal{E}_t\cap E^-) + \mathbb{E}_t(R_t|\overline{\mathcal{E}}_t\cap E^+)\cdot\mathbb{P}_t(\overline{\mathcal{E}}_t\cap E^+). \qquad (44)\end{aligned}$$

The inequality follows since $R_t \ge 0$ a.s. and hence each term above is non-negative. By Lemma 41 and 42, we have

$$\begin{aligned}(44) &\ge \frac{n}{2K}\cdot\mathbb{P}_t(\mathcal{E}_t\cap E_t^-) + \frac{n}{2WK}\cdot\mathbb{P}_t(\overline{\mathcal{E}_t}\cap E_t^+)\\ &\ge \frac{n}{2WK}\cdot\left(\mathbb{P}_t(\mathcal{E}_t)\cdot\mathbb{P}_t(E_t^-) + \mathbb{P}_t(\overline{\mathcal{E}_t})\cdot\mathbb{P}_t(E_t^+)\right)\\ &\ge \frac{n}{2WK}\cdot\left(\frac{1}{2}\cdot\mathbb{P}_t(\mathcal{E}_t) + \frac{1}{W}\cdot\mathbb{P}_t(\overline{\mathcal{E}_t})\right)\\ &\ge \frac{n}{2W^2K},\end{aligned}$$

where in the second inequality we used the key fact that the events $\mathcal{E}_t$ and E_t^+ ($\overline{\mathcal{E}_t}$ and E_t^- resp.) are independent conditional on G_{t-1}. Combining the above with Lemma 40, we obtain

$$\mathbb{E}[R_t] \ge \mathbb{E}[R_t\cdot\mathbb{1}(G_{t-1})] = \mathbb{E}[R_t|G_{t-1}]\cdot\mathbb{P}[G_{t-1}] \ge \frac{n}{2W^2K}\cdot\frac{1}{8} = \frac{n}{16W^2K}.$$

Summing over t and taking the limit, the regret of policy π is bounded as

$$\mathrm{Reg}(\pi) = \overline{\lim_{T\to\infty}}\,\frac{1}{nT}\sum_{t=1}^{T}\mathbb{E}[R_t] \ge \frac{1}{16W^2K},$$

and the proof follows. □

V.2.4. The Second Lower Bound We first present the high level idea. With some foresight define $\delta = n^{-1/2}$ and define an arm a to be δ-*good* if $\mu(a) \ge 1-\delta$ and δ-*bad* otherwise. Since $K = \Omega(\sqrt{n})$, there is a high probability that the optimal arm is also δ-good, and thus we may perform our lower bound analysis conditional on this event.

Fix some t and consider the regret R_{t-W}^{t+W} incurred from time $t-W$ to $t+W$, where we recall that W is viewed as a constant. Consider a policy whose cumulative regret in this period of time is $O(\sqrt{n})$. The key idea is to consider the *cold-start* event that at time t, none of the δ-good arms ever selected is still available. As the name suggests, when this event occurs, in order to attain low regret, the policy has to first identify a δ-good arm and then exploit it. In other words, all past information are "useless" for achieving low regret, and hence any reasonable policy behaves as if the time horizon "restarts" at time t.

Intuitively, if B_t occurs then a high regret is incurred in the near future. More precisely, we will show that $R_{t-W}^{t+W} = \Omega(\sqrt{n})$ conditional on B_t. In fact, since the reward rates are unknown, when selecting unexplored arms (i.e. arms that have never been selected) the policy is essentially randomly guessing. By the uniform reward rate assumption, the policy has to select $\sim \frac{1}{\delta} = \sqrt{n}$ arms before encountering a δ-good arm. Moreover, when a δ-bad arm is selected, an $\Omega(1)$ regret is incurred on average, and therefore $R_{t-W}^{t+W} = \Omega(\sqrt{n})$.

We derive the lower bound by considering the number ℓ of unexplored arms the policy selects during $t-W$ to t. Suppose $\ell > \sqrt{n}$, then by the uniform reward rate assumption, we have $R_{t-W}^{t+W} = \Omega(\sqrt{n})$. In the other case, suppose $\ell \le \sqrt{n}$. Then, due to the uniform reward rate assumption, with $\Omega(1)$ probability none of those ℓ arms is δ-good. Moreover, at time t all δ-good arms that arrived before $t-W$ have expired. Therefore at t, it is likely that no available δ-good arm has ever been selected, and hence B_t occurs, which leads to a high regret. We next formalize the above ideas and establish our $\Omega(n^{-1/2})$ lower bound.

Lemma 43. *Suppose $K > \sqrt{n}$ and define $G_t = \{\mu_{\max}(A_{t-W}^t) \ge 1-\delta\}$ for each $\tau \ge W$. Then,*

$$\mathbb{P}[G_t] \ge \frac{1}{2}.$$

Proof. Since $|\mu_{\max}(A_{t-W}^t)| = WK$ and the reward rate of each arm is drawn i.i.d. from uniform distribution, we have

$$\mathbb{P}[\overline{G_t}] = (1-\delta)^{KW} = (1-\delta)^{\frac{1}{\delta}\cdot KW\delta} \le e^{-KW\delta}.$$

Since $K > \sqrt{n}$, we have $K\delta > n^{\frac{1}{2}} \cdot n^{-\frac{1}{2}} \ge 1$, so $\mathbb{P}[\overline{G_t}] \le e^{-KW\delta} \le e^{-W} \le \frac{1}{2}$, i.e. $\mathbb{P}[G_t \ge \frac{1}{2}]$. □

In particular, this implies a lower bound on the long-run average reward.

Corollary 4. *If $K > \sqrt{n}$, then*

$$\varliminf_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mu^*(a_t) \ge 1-\delta.$$

We will subsequently write $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot|G_t]$ for any t. Fix some t and consider $[t-W, t+W]$. The next lemma says if a large number of unexplored arms is selected, then a high regret is incurred. This is intuitive because the policy has no knowledge about each unexplored arm, except that its reward rate is uniformly distributed.

Lemma 44. *Let $V_\tau = \{|S_\tau| \ge \frac{\sqrt{n}}{4W}\}$ for any $\tau \in \mathbb{N}$, then $\mathbb{E}[R_\tau|V_\tau] \ge \frac{\sqrt{n}}{12W}$.*

Proof. Note that

$$\mathbb{E}_\tau[R_\tau|V_\tau] = \sum_{a\in A} \mathbb{E}_\tau[\pi_\tau(a)\cdot(\mu_t^* - \mu(a))|\ V_\tau] \geq \sum_{a\in A} \mathbb{E}_\tau[\mathbb{1}(a \in S_\tau)\cdot(1-\delta-\mu(a))\mid V_\tau]. \tag{45}$$

Further, since $a \in S_\tau$ is independent of $\mu(a)$ conditional on V_τ, we have

$$\begin{aligned}(45) &= \sum_{a\in A} \mathbb{E}_\tau[\mathbb{1}(a\in S_\tau)|V_\tau]\cdot(1-\delta-\mathbb{E}_\tau\left[\mu(a)|V_\tau\right])\\ &\geq \mathbb{E}\left[|S_\tau|\middle|V_\tau\right]\cdot\left(\frac{1}{2}-\delta\right)\\ &\geq \frac{1}{3\delta} = \frac{\sqrt{n}}{3W},\end{aligned}$$

and the proof follows. □

An arm a is called *unexplored* at the start of round t if it has never been selected before, i.e. $\sum_{\tau=1}^{t-1}\pi_\tau(a) = 0$. Let

$$S_\tau = \{a \in A_{\tau-W}^\tau : \pi_\tau(a) > 0 \text{ and } \sum_{s=1}^{t-1}\pi_s(a) = 0\}$$

be the set of unexplored arms selected by the policy at round τ. The following says if the regret in $[t-W,t]$ is low, then the policy is likely to have under-explored in $[t-W,t]$. Consequently, at time t it is likely that none of the near-optimal arms ever selected is still available.

Lemma 45. *Let* $B_t = \{\mu_{\max}(S_t \cap A_{t-W}^t) \leq 1-\delta\}$. *Suppose* $\mathbb{E}\left[\sum_{\tau=t-W}^t R_\tau\right] \leq \frac{\sqrt{n}}{12W}$, *then* $\mathbb{P}[B_t] \geq \frac{1}{2}$.

Proof. Consider the event $E_\tau = \{\max_{a\in S_\tau}\mu(a) \geq 1-n^{-\frac{1}{2}}\}$. In words, this is the event that one of the unexplored arms selected at time τ has high reward rate. We start with upper bounding $\mathbb{P}[E_\tau]$. Observe that

$$\mathbb{P}[E_\tau] = \mathbb{P}[E_\tau|V_\tau]\cdot\mathbb{P}[V_\tau] + \mathbb{P}[E_\tau|\overline{V_\tau}]\cdot\mathbb{P}[\overline{V_\tau}] \leq \mathbb{P}[V_\tau] + \mathbb{P}[E_\tau|\overline{V_\tau}]. \tag{46}$$

We will bound these two terms separately. First we claim that

$$\mathbb{P}[V_\tau] \leq \frac{1}{4W}. \tag{47}$$

In fact, for a contradiction suppose $\mathbb{P}[V_\tau] > \frac{1}{4W}$, then by Lemma 44,

$$\mathbb{E}[R_\tau] \geq \mathbb{E}[R_\tau|V_\tau]\cdot\mathbb{P}[V_\tau] > \frac{\sqrt{n}}{3}\cdot\frac{1}{4W} = \frac{\sqrt{n}}{12W},$$

contradicting the assumption that $\mathbb{E}\left[\sum_{\tau=t}^{t+W} R_\tau\right] \leq \frac{\sqrt{n}}{12W}$.

To bound the second term in (46). By definition of E_τ, for any integer $C > 0$ we have

$$\mathbb{P}\left[\overline{E}_\tau \middle| S_\tau \le C\right] \ge (1-\delta)^C \ge (1-\delta)^{\frac{1}{\delta}\cdot\delta C} \ge e^{-\delta C},$$

where the last inequality follows since for any $x \in \mathbb{R}$ we have $1 - x \le e^{-x}$. It then follows that

$$\mathbb{P}\left[\overline{E_\tau}\middle|\overline{V_\tau}\right] = \mathbb{P}\left[\overline{E_\tau}\middle| S_\tau \le \frac{\sqrt{n}}{4W}\right] \ge 1 - \delta \cdot \frac{\sqrt{n}}{4W} = 1 - \frac{1}{4W}.$$

Rearranging, we obtain

$$\mathbb{P}\left[E_\tau\middle|\overline{V_\tau}\right] = 1 - \mathbb{P}\left[\overline{E}_\tau\middle|\overline{V_\tau}\right] \le \frac{1}{4W}. \tag{48}$$

Combining (46),(47) and (48), we have $\mathbb{P}[E_\tau] \le \frac{1}{4W} + \frac{1}{4W} = \frac{1}{2W}$.

To conclude the proof, observe that if none of the events E_τ occurs for $\tau \in [t-W, t]$, then B_t occurs. Therefore, by the union bound,

$$\mathbb{P}[B_t] \ge \mathbb{P}\left[\bigcap_{\tau=t-W}^{t} \overline{E_\tau}\right] = 1 - \mathbb{P}\left[\bigcup_{\tau=t-W}^{t} E_\tau\right] \ge 1 - W \cdot \frac{1}{2W} = \frac{1}{2},$$

and the proof follows. □

Intuitively, if B_t occurs and the regret in $[t, t+W]$ is $\le n^{1/2}$, then the policy has to identify a nearly-optimal arm in the next W rounds, which leads to a high regret due to exploration, as formalized below.

Lemma 46 **(Wang et al. (2008)).** $\mathbb{E}\left[\sum_{\tau=t}^{t+W} R_\tau \middle| B_t\right] \ge \frac{\sqrt{n}}{6W}$

We now combine the above to establish the second lower bound (Theorem 30).

Proof of Theorem 30. Decompose R_{t-W}^{t+W} as

$$R_{t-W}^{t+W} = \sum_{\tau=t-W}^{t+W} \mathbb{E}R_\tau = \sum_{\tau=t-W}^{t-1} \mathbb{E}R_\tau + \sum_{\tau=t}^{t+W} \mathbb{E}R_\tau.$$

Suppose the first term is at least $\frac{\sqrt{n}}{12W}$, then the proof follows immediately. Otherwise, by Lemma 45, we have $\mathbb{P}[B_t] \ge \frac{1}{2}$, and hence by Lemma 46 we have

$$\sum_{\tau=t}^{t+W} \mathbb{E}R_\tau \ge \mathbb{E}\left[\sum_{\tau=t}^{t+W} R_\tau \middle| B_t\right] \cdot \mathbb{P}[B_t] \ge \frac{1}{2} \cdot \frac{\sqrt{n}}{6W} = \frac{\sqrt{n}}{12W},$$

and the proof follows. □

V.3. Our Policy and Upper Bound

V.3.1. The Sieve Policy There are three basic ideas for designing policies for bandit problems: Upper Confidence Bound (UCB), Thompson Sampling (TS) and (Explore-Then-Commit) ETC. However, since the arms are arriving online and hence may have different ages, the first two policies would suffer high regret since they do not incorporate the age of the arms. Take UCB as an example. The policy maintains a confidence interval for each arm possibly with different width, and select the arm with the highest upper confidence bound. However, since new arms are arriving in each round, the policy will rarely prefer selecting well-explored over new arms since their confidence intervals are much smaller than the new arms and hence unlikely to have the maximum upper confidence bound.

In contrast, the Explore-Then-Commit (ETC) policy achieves sublinear regret. At each round, the policy uses a small fraction of users to *explore* the newly arrived arms, and assign all remaining users the empirically optimal arm, based on the exploration in the previous steps.

The above ETC policy has an obvious shortcoming: it only explores the newly arrived arms and hence does not make full use of the lifetime W. In particular, this means the policy remains the same for any given lifetime. Consider a natural improvement, dubbed a 2-*Layered Sieve Policy*, specified by two parameters $\varepsilon_0, \varepsilon_1$. When a set A of K arms just arrives, the policy uses $\varepsilon_0 n$ users to explore A by selecting each $a \in A$ for $\frac{\varepsilon_0 n}{K}$ times, and based on these outcomes, the policy computes a subset S of *surviving* arms. In the next round, the policy uses $\varepsilon_1 n$ users to further explore the survivors, and computes an empirically optimal arm. For the remaining $1-\varepsilon_0-\varepsilon_1$ fraction of users, the policy simply selects the empirically optimal arm, among all arms of age at least 2.

More generally, one may repeat this process for $\ell \le W$ times (see Algorithm 12), and hope that the regret bounds decrease as ℓ becomes large. An ℓ-layered Sieve Policy is specified by *exploration intensity* parameters $\varepsilon_0, ..., \varepsilon_{\ell-1}$, where ε_i represents fraction of users the policy uses for (further) exploring the arms that survived the i-th layer of filtering, where $\sum_i \varepsilon_i < 1$.

V.3.2. Analysis of the Sieve Policy The main result in this section is the following regret upper bound for ℓ-layered Sieve policy where $\ell \le W$.

THEOREM 31. *Let $\varepsilon_i = n^{-\frac{\ell-i}{\ell+2}}$ for each $0 \le i \le \ell-1$, then the ℓ-layered Sieve policy satisfies*

$$\operatorname{Reg}(\varepsilon_0, ..., \varepsilon_\ell) \le O\left(\frac{1}{K} + \left(\frac{K \log^2 K}{n}\right)^{\frac{\ell}{\ell+2}}\right).$$

Algorithm 12 Sieve Policy.

```
Input:
    • $\ell$: number of sieve layers,
    • $\varepsilon_1, ...\varepsilon_\ell$: exploration intensities,
    • $n$: number of arms to select in each round
for $t = 0, 1, ...,$ do
    for $i = 0, 1, ..., \ell - 1$ do                                          ▷ Level-$i$ exploration
        if $S^i_{t-i} \neq \emptyset$ then
            $n^t_i \leftarrow \frac{\varepsilon_i n}{|S^i_{t-i}|}$                  ▷ Number of times to select each arm in $S^i_{t-i}$
            for $a \in S^i_{t-i}$ do
                Observe rewards $X^t_{a,1}, ..., X^t_{a,n_i}$
                $\overline{X}^t_a = \frac{1}{n^t_i} \sum_{j=1}^{n^t_i} X^t_{a,j}$                    ▷ Empirical mean
            $\overline{X}^t_{max} \leftarrow \max\{\overline{X}^t_a : a \in S^i_{t-i}\}$            ▷ Empirically maximal reward rate
            $S^{i+1}_{t-i} = \{a \in S^i_{t-i} : |\overline{X}^t_a - \overline{X}^t_{max}| \le 2(n^t_i)^{-1/2}\}$   ▷ Update the surviving arms
    $\hat{A}_t \leftarrow \arg\max\{\overline{X}_a : a \in \bigcup_{\tau=t-W}^{t-\ell} S^\ell_\tau\}$          ▷ Empirically optimal surviving arm
    Select any arm in $\hat{A}_t$ for $n - \sum_{i=0}^{\ell-1} n^t_i$ times                  ▷ Exploitation
```

In particular, when $K \ge \sqrt{n}$, by running the ℓ-layered Sieve policy on $\sqrt{n}$ randomly selected arms from each batch, the upper bound becomes

$$O\left(\frac{1}{\sqrt{n}} + \left(\frac{\sqrt{n}\log^2 n}{n}\right)^{\frac{\ell}{\ell+2}}\right) = O\left(n^{-\frac{\ell}{2\ell+2}}\right).$$

Recall that S^j_t denotes the surviving arms in A_t after j layers. The average regret can then be bounded as

$$\begin{aligned}\mathrm{Reg}(\varepsilon_0, ..., \varepsilon_\ell) \le \varepsilon_0 \cdot 1 + \varepsilon_1 \cdot \left(\mu_{max}(A^t_{t-W}) - \mu_{min}(S^1_{t-1})\right) + \varepsilon_2 \cdot \left(\mu_{max}(A^t_{t-W}) - \mu_{min}(S^2_{t-2})\right) + \\ ... + \varepsilon_{\ell-1} \cdot \left(\mu_{max}(A^t_{t-W}) - \mu_{min}(S^{\ell-1}_{t-\ell+1})\right) + (1 - \varepsilon_0 - \varepsilon_1 - ... - \varepsilon_{\ell-1}) \cdot \left(\mu_{max}(A^t_{t-W}) - \mu(\hat{a}_t)\right),\end{aligned} \tag{49}$$

where $\hat{a}_t$ is the arm for the exploitation users. To bound the above, fix some $j \in [\ell - 1]$ and consider

$$\mu_{max}(A^t_{t-W}) - \mu_{min}(S^j_{t-j}) = \left(\mu_{max}(A^t_{t-W}) - \mu_{max}(A_{t-j})\right) + \left(\mu_{max}(A_{t-j}) - \mu_{min}(S^j_{t-j})\right).$$

Motivated by the above decomposition, we define the *external* regret (ER) and *internal* regret (IR) for A_{t-j} as

$$\mathrm{IR} = \sum_{j=0}^{\ell-1} \varepsilon_j \cdot \left(\mu_{max}(A_{t-j}) - \mu_{min}(S^j_{t-j})\right) + (1 - \varepsilon_0 - \cdots - \varepsilon_{\ell-1}) \cdot \Big(\mu_{max}(A_{t-\ell}) - \mu(\hat{a}_t)\Big), \tag{50}$$

and

$$\begin{aligned}\mathrm{ER} = {}& \varepsilon_1 \cdot \big(\mu_{max}(A_{t-W}^t) - \mu_{max}(A_{t-1})\big) + \varepsilon_2 \cdot \big(\mu_{max}(A_{t-W}^t) - \mu_{max}(A_{t-2})\big) + \cdots \\ & + \varepsilon_{\ell-1} \cdot \big(\mu_{max}(A_{t-W}^t) - \mu_{max}(A_{t-\ell+1})\big) + (1 - \varepsilon_0 - \cdots - \varepsilon_{\ell-1}) \cdot \big(\mu_{max}(A_{t-W}^t) - \mu_{max}(A_{t-\ell})\big)\,. \end{aligned} \tag{51}$$

It is straightforward to verify that $(53) = \mathrm{IR} + \mathrm{ER}$.

Roughly speaking, IR measures the regret due to the estimation error, or more precisely, the difference in reward rates between the worst surviving arm and the optimal arm in a batch of arms. Meanwhile, ER is the regret caused by restricting the choice of the exploitation arm $\hat{a}_t$ to only S_{t-j}^j, i.e. the arms of age at least ℓ.

We bound IR and ER separately. Consider ER first. As an basic fact in probability theory, for m i.i.d. samples $Z_1, ..., Z_m \sim U(0,1)$, it holds $\max_{i\in[m]} Z_i \approx 1 - \frac{1}{m}$. Applying this fact on the available arms A_{t-W}^t at t and substituting $m = WK$, we have $\mu_{max}(A_{t-W}^t) \approx 1 - \frac{1}{WK}$, and hence

Proposition 13. *For any s with $t - W \le s \le t$,*

$$\mathbb{E}\Big[\mu_{max}(A_{t-W}^t) - \mu_{max}(A_s)\Big] \le \frac{1}{K}.$$

The analysis of IR is relatively more involved. Recall that each alive arm is selected for the same number of times for exploration in every layer, and thus we may build a confidence interval around each of them with the same width w_i. Since the reward rates are uniformly drawn, we expect there to be $w_i K$ alive arms, assuming K is large. In the next layer of sieving, the policy uses $\varepsilon_i n$ impressions to explore these $\sim w_i K$ alive arms, so each of these arms will be played for $\frac{\varepsilon_i n}{w_i K}$ times. By concentration bounds, we can bound the width of this new confidence interval as

$$w_{i+1} \sim \left(\frac{\varepsilon_i n}{w_i K}\right)^{-\frac{1}{2}}. \tag{52}$$

To express each w_j in terms of ε_j's, one may expand (52) iteratively and obtain

$$w_{i+1} = w_i^{1/2} \varepsilon_i^{-1/2} \left(\frac{K}{n}\right)^{1/2}.$$

It is then straightforward to verify that for each $i = 1, 2, ... \ell - 1$,

$$w_i \sim \varepsilon_0^{-2^{-i}} \cdot \varepsilon_1^{-2^{-(i-1)}} \cdot ... \cdot \varepsilon_{i-1}^{-\frac{1}{2}} \left(\frac{K}{n}\right)^{1-2^{-i}}.$$

Now we are ready to determine the optimal exploration intensity parameters, i.e. ε_i's, for and ℓ. Recall, by definition of confidence interval, that

$$\mu_{max}(A_{t-j}) - \mu_{min}(S_{t-j}^j) \le w_j$$

for each $j \leq \ell - 1$. Combining with (50), we can bound the internal regret as

$$\mathrm{IR}(\varepsilon_0, \cdots, \varepsilon_{\ell-1}) \leq O\left(\varepsilon_0 + \varepsilon_1 w_1 + \varepsilon_2 w_2 + \cdots + \varepsilon_{\ell-1} w_{\ell-1} + w_\ell\right). \tag{53}$$

To approximately minimize the above, one may select the parameters ε_i's such that

$$\varepsilon_0 = \varepsilon_1 w_1 = \varepsilon_2 w_2 = \cdots = \varepsilon_{\ell-1} w_{\ell-1} = w_\ell. \tag{54}$$

Let $\varepsilon_i = n^{-x_i}$ and $n/K = n^C$, then (54) can be re-written as

$$\begin{aligned}
-x_0 &= -x_1 + \frac{1}{2}x_0 - \frac{1}{2}C \\
-x_0 &= -x_2 + \frac{1}{2}x_1 + \frac{1}{4}x_0 - \frac{3}{4}C \\
-x_0 &= -x_3 + \frac{1}{2}x_2 + \frac{1}{4}x_1 + \frac{1}{8}x_0 - \frac{7}{8}C \\
&\cdots \\
-x_0 &= -x_\ell + \frac{1}{2}x_{\ell-1} + \frac{1}{4}x_{\ell-2} + \ldots + \frac{1}{2^\ell}x_0 - (1 - \frac{1}{2^\ell}) \cdot C \\
-x_0 &= \frac{1}{2}x_\ell + \frac{1}{4}x_{\ell-1} + \ldots + \frac{1}{2^{\ell+1}}x_0,
\end{aligned}$$

which is a linear equation system with $\ell + 1$ variables and equations. One may then verify that the unique solution to the above system is $x_i = \frac{\ell - i}{\ell + 2}C$ for $0 \leq i \leq \ell - 1$, hence the regret is $O\Big((\frac{K}{n})^{\frac{\ell}{\ell+2}}\Big)$.

PROPOSITION 14. *Let $\varepsilon_i = n^{-\frac{\ell-i}{\ell+2}}$ for each $0 \leq i \leq \ell - 1$, then it holds*

$$\mathrm{IR}(\varepsilon_0, \ldots, \varepsilon_\ell) \leq \left(\frac{K \log^2 K}{n}\right)^{\frac{\ell}{\ell+2}}.$$

Combining with Proposition 13, we obtain the following main upper bound in Theorem 31.

V.4. Field Experiment Setup

We further investigate this problem via a field experiment through collaboration with India's largest lockscreen content platform, Glance, that faces exactly this challenge. The platform curates 100-200 *Glance cards* (or simply *cards*) per hour, with content ranging from news articles to short videos. Swiping through cards, users may click through what they find interesting and be redirected by the link for further engagement. Over 70% of the cards expire within 48 hours.

The personalization algorithm that Glance deployed in early 2021 is based on a state-of-the-art Deep Neural Network (DNN) that combines every user's recent interactions with the text and image features of the newly generated cards, to finalize a set of recommended cards sent to the user's mobile device. This DNN-based recommender has been proven to significantly outperform their previous recommender (Oli et al.), such as a Naive Bayes classifier based on the cards' categories.

Despite the efficacy of the current recommender, there is a considerable potential for improvement. In the current DNN recommender, feedback from users only updates their behavioral signature for future prediction, and does not directly leverage the user feedback in making recommendations of content cards. Moreover, it is computationally expensive to re-train the neural network on a regular basis. In fact, it was updated every 12 hours and this already hit their computational limit, which potentially hampered its performance. For example, an underrated card may become unexpectedly popular, but the recommender may only detect this signal 12 hours after the card is released.

In contrast, bandit based policies are computationally less costly and hence may be used incorporate the user feedback to learn the conversion rates in a more timely manner. So the natural question then is, are we able to improve the recommender if we update it more frequently, using ideas from bandit theory?

We answer this question with a resounding yes. We implemented the *simplest* form of our sieve policy in a field experiment on Glance's real system involving 600,000 users over a period of two weeks in mid 2021. Our policy, despite being intentionally handicapped by ignoring personalization, outperformed Glance's current (personalized) recommender significantly in user engagement. In the remainder of this section, we present the details of the setup of the experiment, and explain the experimental results in the Section V.5.

V.4.1. Overview of Glance's System In Glance's current system, the cards are stored in the mobile devices, with expiring cards removed regardless of internet connection. When connected to the internet, the device sends a replenishment request to Glance's system if the current number of cards in the device is lower than a certain threshold, on an hourly basis. Upon receiving such a request, the recommender selects and replenishes the device with the number of cards requested, solely based on a score predicted by the recommender.

We randomly partitioned approximately 600,000 users into two groups: a *control* group where the current DNN-based recommender is deployed, and a *treatment* group where a *variant* of our one-layer sieve policy is implemented.

V.4.2. Treatment Group Policy The policy for the treatment group can be specified by two parameters, the *exploration intensity* ε and *exploration threshold* θ. A card is said to be *well-explored* (resp. *under-explored*) if at least θ impressions have been observed from all users. The policy maintains a posterior distribution for each card, updated hourly using Bayes' rule based on the user-engagement data included in the

Fitting Prior Using NN predictions. For each new card, the policy first computes a suitable prior distribution which we will explain soon.

Upon a replenishment request for r cards from a user, the policy first randomly draws εr under-explored cards. To decide another $(1-\varepsilon)r$ arms, the policy imitates the classic Thompson sampling policy as follows. The policy randomly draws a *score* for each well-explored cards from its posterior distribution, and then selects the top $(1-\varepsilon)r$ cards with highest scores.

In our field experiment, we consider a formulation of the problem where all users are viewed as identical. While this is somewhat impractical, the field experiment results in this setting turned out to be a strong argument *for* our approach: we observed that our online leaning based policy *without* personalization outperforms the offline learning based recommender system *with* personalization.

To formulate Glance's recommendation problem as a multi-armed bandits model, we need to first specify the definition of reward. There are two commonly used metrics that measure the success of recommendation for online platforms: click-through rate (CTR) and duration. For Glance, a click-through occurs if the user clicks on the link embedded in a card. The duration of an impression is simply the amount of time the user spends on it. Intuitively, a long view of on a card suggests the users is likely to be interested, but if the duration is unreasonably long, it is usually because the user is not paying attention to the card (e.g. use phone for flashlight, or forget to turn off the phone). Thus motivated, we define the binary reward of an impression to be 1 if either there is a click through or the duration is between some constants ℓ and u seconds.

One of the most commonly used policy for bandits is Thompson Sampling (TS) (see e.g. the survey of Russo and Van Roy (2016)). The TS policy takes prior distribution as input and maintains a posterior for the reward rates for the arms. Then at each round the policy computes the posterior reward distribution for each arm, then draws a *score* from the posterior and selects the arm with the highest score. TS-based policies has demonstrated excellent performance for newly-created contents. For example, Schwartz et al. (2017) improved the customer acquisition rate by 8% in a live field experiment for displaying ads by deploying a TS-based MAB policy.

Consider the most natural adaptation of the TS policy: suppose a user requests r cards (usually on the magnitude of 10-25) in this hour, then the policy draws a score for each card from its posterior distribution, and recommends the r cards with highest scores to the user. Nonetheless, this policy would select predominantly new cards, and old cards would be rarely selected, even if some of them are shown to be popular by the past interaction data. In fact, on the one hand, due to the sheer size of the user pool, the posterior distributions of cards will quickly collapse to a spike after one or two updates. On the other hand, the posterior distributions of new cards are

Algorithm 13 SetPrior(g).

```
Input: $m > 0$, glance card $g$
Output: $\alpha, \beta$                                  ▷ Return a Beta distribution
Randomly sample $m$ users $u_1, ..., u_m$
for $i = 1, ..., m$ do
    $\mu(u_i, g) \leftarrow$ neural network prediction for pair $(u_i, g)$
Compute the sample mean $\bar{x}$ and variance $\bar{v}$:
    $$\bar{x} \leftarrow \frac{1}{m}\sum_{i=1}^{m} \mu(u_i, g), \quad \bar{v} \leftarrow \frac{1}{m-1}\sum_{i=1}^{m} (\mu(u_i, g) - \bar{x})^2$$
Set up Beta prior using the method of moments:
    $$\alpha \leftarrow \bar{x}\left(\frac{\bar{x}(1-\bar{x})}{\bar{v}} - 1\right), \quad \beta = \frac{1-\bar{x}}{\bar{x}}\alpha$$
```

more flat, so they are much more likely to be assigned an extremely high score by the TS policy and hence be recommended to the user.

V.4.3. Counterfactual Simulation We performed an counterfactual simulation using Glance's interaction data in February 2021, aimed at demonstrating the effectiveness of the one-layer Sieve policy. The data includes the following information about each interaction:

1. the hashed user name,
2. the card name,
3. the duration,
4. whether a click-through occurred
5. the start time of this interaction.

As the key challenge for the simulation, for each pair of user and card that had no interaction, we do not know what *could have* happened when this card was assigned to the user. To circumvent this issue, we consider the following counterfactual simulation. First we sort the interaction data according to their start time, and partitioned the sorted data into *blocks* of size 50, which we will consider one by one. Since the click-through (CT) events are rare (0.5%), to better compare the performances between policies, for each we define interaction the *reward* to be 1 if either the duration reaches 0.5 seconds, or there is a click-through. For each block, we let the policy under consideration select the 20% cards (i.e. 10 out of 50) using its selection criterion. At the end of the simulation, we compute the average reward rate of the cards selected by the policy.

Algorithm 14 Randomized One-Layer Sieve Policy

```
Input:$\varepsilon \in [0,1], \theta \geq 0$
for each hour $t = 1, 2, ...$ do
    Receive a set $A_{new}$ of new cards
    Update the set $A$ of available cards
    for each card $g \in A$ do
        if $g \in A_{new}$ then
            $(\alpha_g, \beta_g) \leftarrow \mathrm{SetPrior}(g)$.    ▷ Compute a prior reward distribution
        else
            $n_g \leftarrow$ number of interactions of $g$ in the last hour
            $h_g \leftarrow$ number of conversions of $g$ in the last hour
            $\alpha_g \leftarrow \alpha_g + h_g,\ \beta_g \leftarrow \beta_g + n_g - h_g$    ▷ Update the posterior distribution
    $A_{\text{well}} \leftarrow \{g : \alpha_g + \beta_g > \theta\}$    ▷ Well-explored cards
    for each user $u$ do    ▷ Thompson Sampling
        Receive the number $r_u$ of cards requested by $u$
        $A_u \leftarrow$ cards that have never been assigned to $u$
        for each card $g \in A_u$ do    ▷ Sample from the posterior
            Draw $X_{u,g} \sim \mathrm{Beta}(\alpha_g, \beta_g)$
        Sort $A_u \bigcap A_{\text{well}}$ by $X_{u,g}$ in non-increasing order as $g_1, g_2, ...$
        Sort $A_u \backslash A_{\text{well}}$ by $X_{u,g}$ in non-increasing order as $g'_1, g'_2, ...$
        $i, i' \leftarrow 1$
        for $j = 1, ...r_u$ do    ▷ Select and rank the cards
            $Z_j \leftarrow Ber(\varepsilon)$    ▷ Decide whether to explore or exploit
            if $Z_j = 1$ then    ▷ Exploit
                $S_j \leftarrow g_i$
                $i \leftarrow i + 1$
            else    ▷ Explore
                $S_j \leftarrow g'_{i'}$
                $i' \leftarrow i' + 1$
        Send to $u$ the cards $\{S_j : j = 1, ..., r_u\}$
```

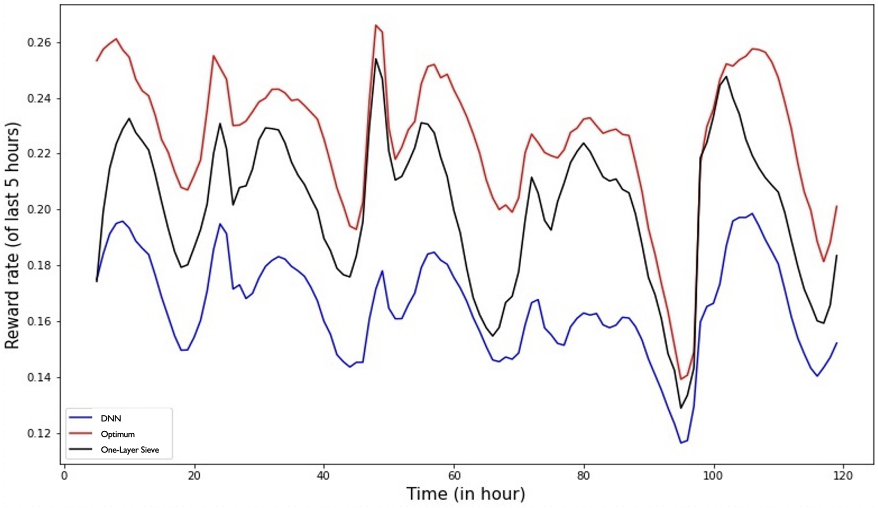

Figure 11 Counterfactual Simulation

We examine two policies. First we consider the Optimal Policy, which computes the average reward rate of each card in advance, and then for each block selects the top 20% cards with highest reward rates. The second policy is our One-Layer Sieve Policy, with exploration intensity chosen to be $\varepsilon_0 = 0.2$. More precisely, in each block, on the exploration side, we select $0.2 \times 10 = 2$ newly-arriving (i.e. released within 1 hour) cards randomly, and in case there are less than 2 such cards, we select all of them. On the exploitation side, we select $(1 - 0.2) \times 10 = 8$ cards with the highest reward rates in the past interactions.

We compare the average reward rates of the selected cards in our Sieve policy with that of the Optimal Policy, as well as the overall average reward rate over all interactions, as shown in Figure 11. Our Sieve Policy outperformed the overall reward rate by 20.8%.

V.4.4. Integrating Offline and Online Learning As one of the merits compared to other optimism-based policies, the TS policy is able to incorporate the prior information, which in our case, is the reward prediction returned by the currently deployed DNN. As Russo et al. (2017) pointed out, “a careful choice of prior can significantly improve learning performance”. A good prior becomes even more crucial in the face of short lifetime of arms. In our problem, however time horizon (i.e. lifetime) of each card is short, hence an inaccurate prior (or not having a prior at all) slows down the learning rate hence spoils the performance of the recommender. In contrast, the prior is not as important for long-lived bandits since the time wasted for correcting the poorly-chosen prior is negligible.

Thus motivated, we fit a Beta prior for each card using the DNN-based predictions on the users. We first randomly sample 500 users, and then predict their reward rates for this card by

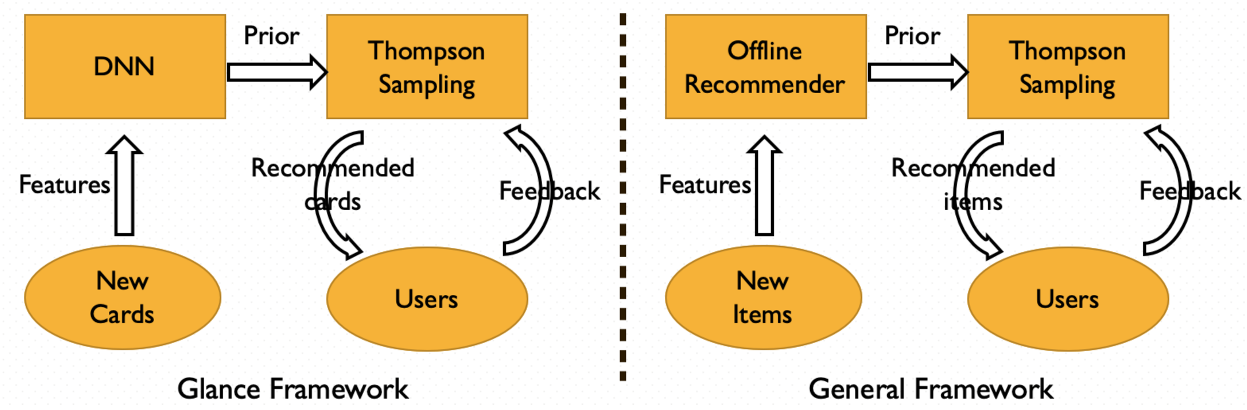

Figure 12 Generalizing our method to other applications.

querying the DNN-based model trained beforehand. Denote the sample mean and variance as $\bar{x}, \bar{v}$, then the methods of moments (see e.g. Wasserman (2006)) estimates the Beta parameters $\hat{\alpha}, \hat{\beta}$ as

$$\hat{\alpha} = \bar{x}\left(\frac{\bar{x}(1-\bar{x})}{\bar{v}} - 1\right), \quad \hat{\beta} = \frac{1-\bar{x}}{\bar{x}}\hat{\alpha}.$$

Our framework, summarized as Figure 2, can be generalized in many ways to various other applications. First, the prior distribution can be replaced with any reasonable offline predictor, not necessarily a neural network based one. Second, the items recommended can be any short-lived items, ranging from videos or news articles for online platforms, to new products in brick-and-mortar retailing.

V.5. Field Experiment Results

We implemented our Randomized One-Layer Sieve Policy (see Algorithm 12) in Glance's real system in the first 14 days of July 2021, and for the sake of comparison, we also requested Glance to extract the impression data from the first 14 days of May 2021, on the same subset of users.

Before analyzing the experimental results, we first handle the outliers in the data, which are introduced mainly in two ways. In practice, users may accidentally swipe two cards in a row, without even looking at the second card. Alternatively, users may be distracted by other activities and leave their phones unattended for minutes, generating extremely high duration.

We filter the interaction data as follows. On the one hand, since most cards are either articles or short videos under 5 minutes, we remove impressions with duration over $u = 300$ seconds. On the other hand, we also remove the impressions with duration less than $\ell = 0.2$ seconds.

V.5.1. Metrics We will analyze the user engagement on two levels: *per-user-per-day* and *per-impression*. Moreover, we consider two natural metrics (duration and click-throughs) for user engagement, hence giving *four* metrics in total, as shown in Table 11.

In the per-impression analysis, we treat each impression as an individual, independent sample. For example, the per-impression mean duration in Table 12 is simply the ratio between the total duration and the total number of impressions. Since our Sieve Policy also performs online learning on the per-impression level, we anticipate the MAB group to outperform in terms of both mean duration and Click-Through Rate (CTR), assuming the experiment is implemented correctly.

Table 11 Types of Data In the Analysis

	CT	Duration
Per User-Day	integral	numeric
Per Impression	binary	numeric

However, Glance's ultimate goal is to improve the *total* user engagement, rather than the *per-impression* engagement, which motivated our analysis on the per-user-per-day level. Before providing the formal definition, we first explain why it is necessary to group the data by users and days respectively. As we recall, the partition of control versus experimental group was randomly decided by Glance much earlier than the field experiment. In particular, a portion of old users have left the platform, and thus the control and treatment groups have evolved to different sizes over time. Therefore it is reasonable not to simply aggregate the user engagement in the two groups.

On the other side, at first sight it seems reasonable to consider the total duration of a fixed user over *all* days. This metric, however, is flawed. In fact, our recommendation has little impact on users' decisions to enter the Glance app. Rather, the frequency at which users enter the app is affected by many external factors, such as holidays and weekend, which may potentially introduce extra noise. We thus also group each user's impression data by day as follows. For each user-day pair (u, d), we sum over the duration of all impressions of u on day d, hence obtaining a tuple (u, d, D_{ud}), where D_{ud} denotes the total duration. Moreover, we only consider the days d when u has at least one impression, in other words, we only consider days when the user actually used the app.

Organization. In Section V.5.2 we will perform the analysis for all users, and then in Section V.5.3 we consider a subset of users who were more engaged, which we call *engaged* users. For each of these two metrics, we will apply two fundamental statistical methods, the significance test and difference-in-difference (DID) test.

Table 12 Overall Statistics for All Users

			May		July	
			NN	MAB	NN	MAB
Per User-Day	Duration	Mean	175.910	175.548	137.059	142.618
		SE Mean	0.699	0.659	0.6081	0.597
		Median	44.250	44.279	32.973	34.430
	#CT	Mean	1.275	1.273	0.941	1.010
		SE Mean	9.251e-03	8.814e-03	7.276e-03	7.549e-03
Per Impression	Duration	Mean	3.9697	4.0195	4.1183	4.2391
		SE Mean	4.529e-03	4.402e-03	5.738e-03	5.599e-03
		Median	0.693	0.697	0.702	0.703
	CTR	Mean	2.887e-02	2.915e-02	2.827e-02	3.001e-02
		SE Mean	4.698e-05	4.568e-05	5.804e-05	5.671e-05

Table 13 Significance Testing

		Basic		Bootstrap	
		Z-score	p-value	Z-score	p-value
Per User-Day	Duration	4.610	2.018e-06	4.6197	1.921e-06
	CT	4.259	1.027e-05	4.2556	1.042e-05
Per Impression	Duration	6.963	1.665e-12	6.972	1.556e-12
	CT	12.999	6.127e-39	12.933	1.469e-38

V.5.2. Analysis For All Users In this section we consider the engagement of all users in terms of the four metrics in Table 11. The unit of duration in our our tables is second. We first consider the overall statistics as summarized in Table 12. We observe that in May the user engagement of the two groups are approximately identical, but in July the MAB group has a significantly higher mean user engagement. Moreover, such improvement also appeared in *median* duration, indicating that such improvement is more likely to be caused by an overall inflation in duration, rather than just a heavier tail in the distribution.

It is worth noting that the user engagement per user per day decreased from May to July. This is because May 2021 was when the Covid-19 pandemic reached its peak in India, where most Glance users are located. During the pandemic lockdown, the users may have had more time to spend on the app, resulting in a higher total engagement.

In the remainder of this subsection, we will show the statistical significance of our improvement using two approaches: significance testing and difference-in-differences (DID) regression.

Significance Test. Suppose the true distribution of the metric of interest (e.g. duration or click-thru) are X_{May}, X_{July} for the NN group and Y_{May}, Y_{July} for the MAB group. For each of these two metrics, we are interested in the *difference-in-differences* before and after the bandit policy was deployed, i.e. $\Delta = (Y^{July} - X^{July}) - (Y^{May} - X^{May})$. We aim to test between the hypotheses

$$H_0 : E[\Delta] \leq 0 \quad \text{vs.} \quad H_1 : E[\Delta] > 0.$$

For $m \in$ {May, July}, let $\overline{X}^m$,$\overline{Y}^m$ to be the sample mean in month m, possibly over different number of samples. We first consider the basic Z-score, defined as

$$Z = \frac{\left(\overline{Y}^{July} - \overline{X}^{July}\right) - \left(\overline{Y}^{May} - \overline{X}^{May}\right)}{\hat{S}} \tag{55}$$

where

$$\begin{aligned} \hat{S} &= SE\left(\left(\overline{Y}^{July} - \overline{X}^{July}\right) - \left(\overline{Y}^{May} - \overline{X}^{May}\right)\right) \\ &= \sqrt{Var\left(\left(\overline{Y}^{July} - \overline{X}^{July}\right) - \left(\overline{Y}^{May} - \overline{X}^{May}\right)\right)}. \end{aligned}$$

Assuming the samples are nearly independent, we may approximate the above as

$$\sqrt{\frac{1}{N_X^{May}} S^2_{X^{May}} + \frac{1}{N_X^{July}} S^2_{X^{July}} + \frac{1}{N_Y^{May}} S^2_{Y^{May}} + \frac{1}{N_Y^{July}} S^2_{Y^{July}}},$$

where S_Z^2 is the sample variance of a pool Z of samples. As summarized in the "Basic" column of Table 13, we reject the null hypothesis that the treatment effect is insignificant.

However, in reality the samples are not independent, since each user may (1) appear in both months, (2) have multiple data points in a month, and (3) the same set of glance cards are shown to both the treatment and control group. In fact, a user may have at most 14 data points in each group since the experiment lasted for 14 days. We remove the dependence by bootstrapping as follows. From each of these four pools of data points, we randomly draw 10^6 samples with replacement, and redefine each $\bar{Z}$ in (55) for each of $Z = X^{May}, X^{July}, Y^{May}, Y^{July}$ to be the bootstrap sample mean. The results with bootstrapping is consistent with our earlier findings, as illustrated in the "Bootstrap" column in Table 12.

Difference-In-Differences Regression. We first illustrate DID regression for per-user-per-day user engagement. To this aim, we vectorize each tuple (u, d, Y_{ud}) into a vector $(t_{ud}, i_{ud}, t_{ud} \cdot i_{ud}, Y_{ud})$ where

$$t_{ud} = \mathbb{1}[\text{day } d \text{ is in July}] \quad \text{and} \quad i_{ud} = \mathbb{1}[\text{user } u \text{ is in MAB group}]$$

denote the time and intervention indicators respectively, and $Y_{ud} \in \{C_{ud}, D_{ud}\}$ is the metric under consideration (i.e. click-throughs or duration of user u on day d).

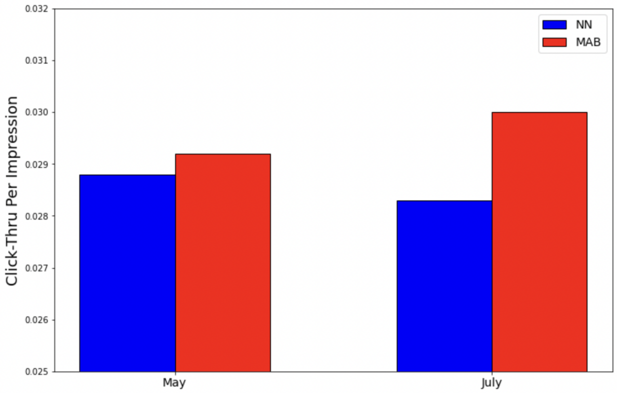

Figure 13 Click-through per impression.

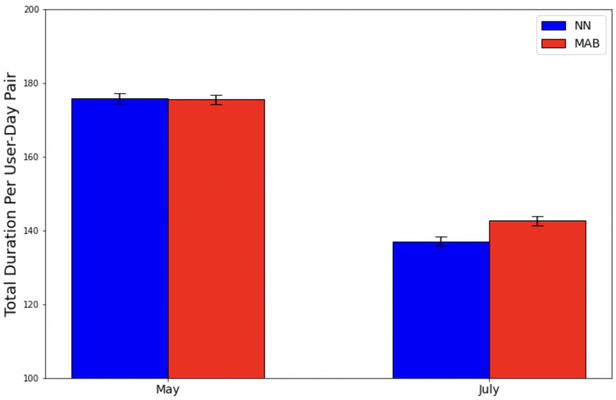

Figure 14 Duration per user-day pair.

We assume the metric Y_{ud} follows the linear model

$$Y_{ud} = \beta_0 + \beta_1 t_{ud} + \beta_2 i_{ud} + \beta_3 t_{ud} i_{ud} + \varepsilon_{ud} \tag{56}$$

where $\varepsilon_{ud} \sim N(0, \sigma^2)$ with unknown variance σ^2. Intuitively, β_1 measures the effect of being assigned to the treatment group, and β_2 captures the overall trend over time. Thus, if there is no treatment effect, the differences between the two groups should remain unchanged across May and July, and therefore the means of the samples from the four pools (shown as the three solid red dots and one hollow dot in Figure 15) will form a perfect parallelogram.

Now suppose there is indeed a positive treatment effect, then the top-right corner of this quadrilateral will be raised (shown as the highest solid red dot in Figure 15). The variable β_3 for the composite variable measures exactly this lift. In fact, for day d in July and user u in MAB group, we have $i_{ud} = t_{ud} = 1$, we have $\mathbb{E}Y_{ud} = \beta_0 + \beta_1 + \beta_2 + \beta_3$, which is higher than the hollow dot by β_3. Finally, one can easily verify that β_0 is simply the mean engagement of control group users in May, by setting $t_{ud} = i_{ud} = 0$.

Under the Gaussian noise assumption, we are able to compute confidence intervals and p-values for the coefficients β_i's, as shown in Tables 14. For both duration and CT, the coefficients β_3 are positive, with very low p-values, therefore the treatment effect (i.e. whether or not a user is assigned to the MAB group) is indeed significant. Meanwhile, the coefficients β_2 for the intervention variables have high p-values, confirming that the partition of users is sufficiently random, at least on the per-user-per-day level.

As shown in the second half of Table 14, we also consider per-impression user engagement. Similar to the above analysis, we convert each impression j into a three-dimensional binary vector (t_j, i_j, Y_j) where Y_j is either the duration (continuous) or click-through indicator (binary) for

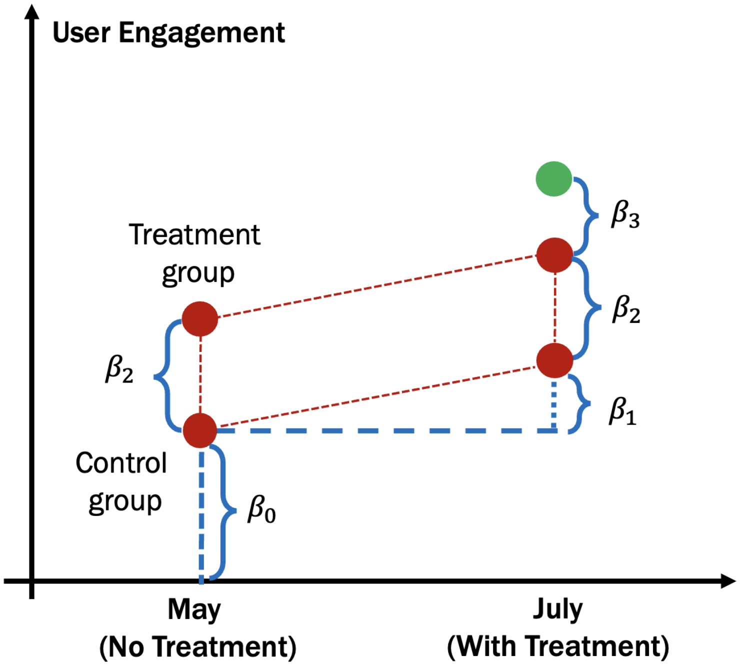

Figure 15 **Illustration of DID regression.**

impression j. While the duration per impression can be analyzed using the linear regression model (56), for per impression click-through the labels Y_j are *binary* and hence we apply logistic regression instead: we assume $Y_j \sim \mathrm{Ber}\left(\frac{e^z}{1+e^z}\right)$ where

$$z = \beta_0 + \beta_1 t_j + \beta_2 i_j + \beta_3 t_j i_j.$$

As opposed to the per-user-per-day regression, in this case *all* coefficients have tiny p-values, for both CT and duration. In particular, the coefficient β_2 for intervention has low p-value, indicating that the initial user-partition may not be truly random, in terms of per impression engagement. Nonetheless, this difference is interpretable. In fact, our experiment was performed on random user-groups that Glance has been using for months prior to our field test, on which some previous experiments have been performed, potentially causing this discrepancy in user behavior.

V.5.3. Analysis For Engaged Users In this subsection we analyze the engagement of special subsets of "engaged" users. We first consider the retention rate from May to July. A user is said to be *engaged* in month $m \in \{\text{May, July}\}$ if she has at least k click-throughs in month m. The attrition rate (for NN and MAB group respectively) is then defined to be the proportion of engaged users in May who remained engaged in July. While there is no obvious choice for the threshold k, Figure 16 shows that for each $k = 1, 2, ..., 10$, the attrition rates of the MAB group is consistently higher than

Table 14 Difference-In-Differences Regression

			Coef.	Std. Dev.	t	p-value	0.025Q	0.975Q
Per User-Day	Duration	β_0	175.9103	0.640	274.941	0.000	174.656	177.164
		β_1	-38.8514	0.942	-41.263	0.000	-40.697	-37.006
		β_2	-0.3622	0.887	-0.409	0.683	-2.100	1.375
		β_3	**5.9208**	1.303	4.544	**2.759e-06**	3.367	8.475
	#CT	β_0	1.2750	0.008	153.851	0.000	1.259	1.291
		β_1	-0.3341	0.012	-27.394	1.616e-165	-0.358	-0.310
		β_2	-0.0016	0.011	-0.141	0.888	-0.024	0.021
		β_3	**0.0704**	0.017	4.171	**1.516e-05**	0.037	0.103
Per Impression	Duration	β_0	3.9697	0.005	863.796	0.000	3.961	3.979
		β_1	0.1486	0.007	20.234	2.753e-89	0.134	0.163
		β_2	0.0497	0.006	7.781	3.597e-15	0.037	0.062
		β_3	**0.0711**	0.010	6.998	**1.298e-12**	0.051	0.091
	CTR	β_0	-3.5198	0.002	-2092.794	0.000	-3.523	-3.517
		β_1	-0.0161	0.003	-5.947	1.365e-09	-0.021	-0.011
		β_2	0.0133	0.002	5.712	5.582e-09	0.009	0.018
		β_3	**0.0474**	0.004	12.819	**6.417e-38**	0.040	0.055

Note: All regression are linear regression except for per impression CT, where we applied logistic regression due to binary labels.

the NN group, which suggests that the choice of k is likely not essential. Thus, in the remaining analysis we will fix this threshold to be $k = 4$ and repeat the analysis in Section V.5.2.

By comparing Table 12 and 15, we observed that for engaged users, both the per-user-per-day duration and click-throughs are notably higher than the average over all users, indicating that our definition for “engaged” user indeed captures the enthusiasm of users. As another noteworthy observation, for each of the four DID regressions in Table 14 and 17, the coefficient β_3 for the magnitude of the composite variable is higher for the engaged users than for all users. This suggests that the treatment effect is even more significant for the engaged users.

Table 15 Overall Statistics For Engaged Users

			May		July	
			NN	MAB	NN	MAB
Per User-Day	Duration	Mean	404.707	403.228	303.002	314.272
		SE Mean	2.142	2.010	2.670	2.496
		Median	196.970	199.845	133.348	146.230
	#CT	Mean	4.235	4.236	2.657	2.866
		SE Mean	3.241e-02	3.080e-02	3.642e-0s2	3.661e-02
Per Impression	Duration	Mean	3.790	3.866	3.659	3.850
		SE Mean	5.922e-03	5.802e-03	1.022e-02	1.076e-02
		Median	0.619	0.622	0.594	0.598
	CTR	Mean	3.954e-02	4.043e-02	3.198e-02	3.498e-02
		SE Mean	6.802e-05	6.671e-05	1.074e-05	1.062e-05

Table 16 Significance Testing For Engaged Users

		Basic		Bootstrap	
		Z-score	p-value	Z-score	p-value
Per User-Day	Duration	2.719	3.273e-03	2.717	3.289e-03
	#CT	3.056	1.121e-02	3.064	1.089e-02
Per Impression	Duration	6.996	1.321e-12	7.060	8.301e-13
	CTR	11.626	1.513e-31	11.478	8.482e-31

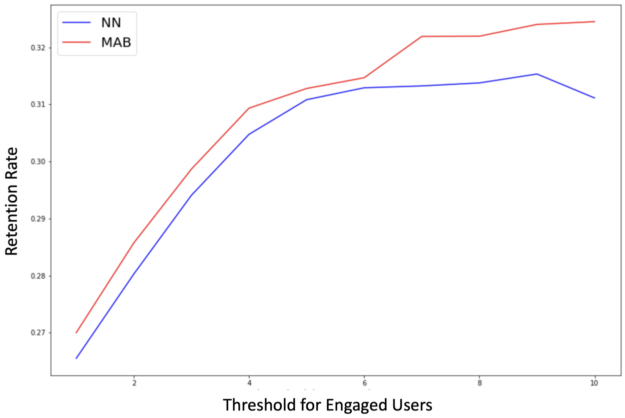

Figure 16 Attrition rates of the MAB and NN group.

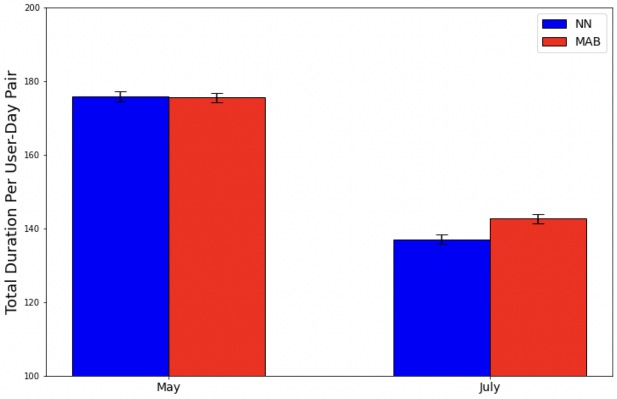

Figure 17 Duration per user-day pair.

References

Micah Adler and Brent Heeringa. Approximating optimal binary decision trees. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 1–9. Springer, 2008.

Arpit Agarwal, Shivani Agarwal, Sepehr Assadi, and Sanjeev Khanna. Learning with limited rounds of adaptivity: Coin tossing, multi-armed bandits, and ranking from pairwise comparisons. In *Conference on Learning Theory*, pages 39–75. PMLR, 2017.

Table 17 Difference-In-Differences Regression for Engaged Users

			Coef.	Std. Dev.	t	p-value	0.025Q	0.975Q
Per User-Day	Duration	β_0	404.7074	2.008	201.569	0.000	400.772	408.643
		β_1	-101.7057	3.692	-27.548	2.338e-167	-108.942	-94.470
		β_2	-1.4791	2.782	-0.532	0.595	-6.932	3.974
		β_3	12.7489	5.083	2.508	0.012	2.786	22.711
	CT	β_0	4.2353	0.030	140.602	0.000	4.176	4.294
		β_1	-1.5787	0.055	-28.502	5.532e-179	-1.687	-1.470
		β_2	0.0006	0.042	0.015	0.988	-0.081	0.082
		β_3	0.2086	0.076	2.736	0.006	0.059	0.358
Per Impression	Duration	β_0	3.7789	0.006	634.095	0.000	3.767	3.791
		β_1	-0.131	0.012	-10.854	9.544e-28	-0.154	-0.107
		β_2	0.0723	0.008	8.708	1.546e-18	0.056	0.089
		β_3	0.1162	0.017	6.998	1.298e-12	0.084	0.149
	CT	β_0	-3.190	0.002	-1771.359	0.000	-3.193	-3.186
		β_1	-0.220	0.004	-55.949	0.000	-0.228	-0.212
		β_2	0.0237	0.002	9.500	1.049e-21	0.019	0.029
		β_3	0.0691	0.005	12.942	1.303e-38	0.059	0.080

Note: As in the previous section, all regression are linear regression except for per impression CT, where we applied logistic regression due to binary labels.

Charu C Aggarwal et al. *Recommender systems*, volume 1. Springer, 2016.

Rajeev Agrawal. The continuum-armed bandit problem. *SIAM journal on control and optimization*, 33(6): 1926–1951, 1995.

Esther M Arkin, Henk Meijer, Joseph SB Mitchell, David Rappaport, and Steven S Skiena. Decision trees for geometric models. *International Journal of Computational Geometry & Applications*, 8(03):343–363, 1998.

Peter Auer, Ronald Ortner, and Csaba Szepesvári. Improved rates for the stochastic continuum-armed bandit problem. In *International Conference on Computational Learning Theory*, pages 454–468. Springer, 2007.

Yossi Aviv and Gustavo Vulcano. Dynamic list pricing. In *The Oxford Handbook of Pricing Management.* 2012.

Yossi Azar and Iftah Gamzu. Ranking with submodular valuations. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms (SODA'11)*, pages 1070–1079. SIAM, 2011.

Moshe Babaioff, Shaddin Dughmi, Robert Kleinberg, and Aleksandrs Slivkins. Dynamic pricing with limited supply. *ACM Transactions on Economics and Computation (TEAC)*, 3(1):1–26, 2015.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 207–216. IEEE, 2013.

Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML '06), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, pages 65–72, 2006.

Hamsa Bastani, David Simchi-Levi, and Ruihao Zhu. Meta dynamic pricing: Learning across experiments. *Available at SSRN 3334629*, 2019.

Gowtham Bellala, Suresh K. Bhavnani, and Clayton Scott. Active diagnosis under persistent noise with unknown noise distribution: A rank-based approach. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, pages 155–163, 2011.

Omar Besbes and Assaf Zeevi. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research*, 57(6):1407–1420, 2009.

John R Birge, Hongfan Chen, and N Bora Keskin. Markdown policies for demand learning with forward-looking customers. *Available at SSRN 3299819*, 2019.

Gabriel R Bitran and Susana V Mondschein. Periodic pricing of seasonal products in retailing. *Management science*, 43(1):64–79, 1997.

Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. *arXiv preprint arXiv:1904.10040*, 2019.

Tamer Boyacı and Özalp Özer. Information acquisition for capacity planning via pricing and advance selling: When to stop and act? *Operations Research*, 58(5):1328–1349, 2010.

Josef Broder. Online algorithms for revenue management. 2011.

Josef Broder and Paat Rusmevichientong. Dynamic pricing under a general parametric choice model. *Operations Research*, 60(4):965–980, 2012.

Sabri Celik, Alp Muharremoglu, and Sergei Savin. Revenue management with costly price adjustments. *Operations research*, 57(5):1206–1219, 2009.

Venkatesan T Chakaravarthy, Vinayaka Pandit, Sambuddha Roy, and Yogish Sabharwal. Approximating decision trees with multiway branches. In *International Colloquium on Automata, Languages, and Programming*, pages 210–221. Springer, 2009.

Venkatesan T. Chakaravarthy, Vinayaka Pandit, Sambuddha Roy, Pranjal Awasthi, and Mukesh K. Mohania. Decision trees for entity identification: Approximation algorithms and hardness results. *ACM Trans. Algorithms*, 7(2):15:1–15:22, 2011.

Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. Mortal multi-armed bandits. *Advances in neural information processing systems*, 21:273–280, 2008.

Boxiao Chen, Xiuli Chao, and Yining Wang. Data-based dynamic pricing and inventory control with censored demand and limited price changes. *Operations Research*, 68(5):1445–1456, 2020.

Ningyuan Chen. Multi-armed bandit requiring monotone arm sequences. *arXiv preprint arXiv:2106.03790*, 2021.

Yiwei Chen and Vivek F Farias. Robust dynamic pricing with strategic customers. *Mathematics of Operations Research*, 43(4):1119–1142, 2018.

Yuxin Chen, Seyed Hamed Hassani, and Andreas Krause. Near-optimal bayesian active learning with correlated and noisy tests. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, pages 223–231, 2017.

Wang Chi Cheung, David Simchi-Levi, and He Wang. Dynamic pricing and demand learning with limited price experimentation. *Operations Research*, 65(6):1722–1731, 2017.

Ferdinando Cicalese, Eduardo Sany Laber, and Aline Medeiros Saettler. Diagnosis determination: decision trees optimizing simultaneously worst and expected testing cost. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 414–422, 2014.

Richard Combes and Alexandre Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. In *International Conference on Machine Learning*, pages 521–529, 2014.

Eric W Cope. Regret and convergence bounds for a class of continuum-armed bandit problems. *IEEE Transactions on Automatic Control*, 54(6):1243–1253, 2009.

Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, pages 191–198, 2016.

Sanjoy Dasgupta. Analysis of a greedy active learning strategy. In *Advances in neural information processing systems*, pages 337–344, 2005.

Arnoud den Boer and N Bora Keskin. Dynamic pricing with demand learning and reference effects. *Available at SSRN 3092745*, 2020.

Arnoud V den Boer and Bert Zwart. Simultaneously learning and optimizing using controlled variance pricing. *Management science*, 60(3):770–783, 2013.

Arnoud V den Boer and Bert Zwart. Dynamic pricing and learning with finite inventories. *Operations research*, 63(4):965–978, 2015.

Utpal M Dholakia. If you are going to raise prices, tell customers why. *Harvard Business Review*, 2021.

Wedad Elmaghraby and Pınar Keskinocak. Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management science*, 49(10):1287–1309, 2003.

Vivek F Farias and Benjamin Van Roy. Dynamic pricing with a prior on market response. *Operations Research*, 58(1):16–29, 2010.

Kris Johnson Ferreira, David Simchi-Levi, and He Wang. Online network revenue management using thompson sampling. *Operations research*, 66(6):1586–1602, 2018.

Guillermo Gallego and Garrett Van Ryzin. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management science*, 40(8):999–1020, 1994.

Kyra Gan, Su Jia, Andrew Li, and Sridhar R Tayur. Toward a liquid biopsy: Greedy approximation algorithmsfor active sequential hypothesis testing. *Available at SSRN*, 2021.

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. *arXiv preprint arXiv:1904.01763*, 2019.

M.R. Garey and R.L. Graham. Performance bounds on the splitting algorithm for binary testing. *Acta Informatica*, 3:347–355, 1974.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Intell. Res.*, 42:427–486, 2011. doi: 10.1613/jair.3278. URL `https://doi.org/10.1613/jair.3278`.

Daniel Golovin and Andreas Krause. Adaptive submodularity: A new approach to active learning and stochastic optimization. *CoRR*, abs/1003.3967, 2017. URL `http://arxiv.org/abs/1003.3967`.

Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal bayesian active learning with noisy observations. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010 (NIPS'10), Vancouver, British Columbia, Canada.*, pages 766–774, 2010.

Google. Transforming specialty retail with ai. Technical report, 2021.

Andrew Guillory and Jeff A. Bilmes. Average-case active learning with costs. In *Algorithmic Learning Theory, 20th International Conference, ALT 2009, Porto, Portugal, October 3-5, 2009. Proceedings*, pages 141–155, 2009.

Andrew Guillory and Jeff A. Bilmes. Interactive submodular set cover. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 415–422, 2010.

Andrew Guillory and Jeff A. Bilmes. Simultaneous learning and covering with adversarial noise. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 369–376, 2011.

Anupam Gupta, Viswanath Nagarajan, and R Ravi. Approximation algorithms for optimal decision trees and adaptive tsp problems. *Mathematics of Operations Research*, 42(3):876–896, 2017.

Swati Gupta and Vijay Kamble. Individual fairness in hindsight. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 805–806, 2019.

Steve Hanneke. A bound on the label complexity of agnostic active learning. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML '07), Corvallis, Oregon, USA, June 20-24, 2007*, pages 353–360, 2007.

J Michael Harrison, N Bora Keskin, and Assaf Zeevi. Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, 58(3):570–586, 2012.

Aliza Heching, Guillermo Gallego, and Garrett van Ryzin. Mark-down pricing: An empirical analysis of policies and revenue potential at one apparel retailer. *Journal of revenue and pricing management*, 1 (2):139–160, 2002.

Paul Heidhues and Botond Kőszegi. Regular prices and sales. *Theoretical Economics*, 9(1):217–251, 2014.

Christian Homburg, Wayne D Hoyer, and Nicole Koschate. Customers' reactions to price increases: do customer satisfaction and perceived motive fairness matter? *Journal of the Academy of Marketing Science*, 33(1):36–49, 2005.

Zhenyu Hu, Xin Chen, and Peng Hu. Dynamic pricing with gain-seeking reference price effects. *Operations Research*, 64(1):150–157, 2016.

Laurent Hyafil and Ronald L. Rivest. Constructing optimal binary decision trees is Np-complete. *Information Processing Letters*, 5(1):15–17, 1976/77.

Sungjin Im, Viswanath Nagarajan, and Ruben Van Der Zwaan. Minimum latency submodular cover. *ACM Transactions on Algorithms (TALG)*, 13(1):13, 2016.

Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, Drew Bagnell, and Siddhartha S. Srinivasa. Near optimal bayesian active learning for decision making. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, pages 430–438, 2014.

Su Jia, Viswanath Nagarajan, Fatemeh Navidi, and R. Ravi. Optimal decision tree with noisy outcomes. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 3298–3308, 2019.

Su Jia, Andrew Li, and R Ravi. Markdown pricing under unknown demand. *Available at SSRN 3861379*, 2021.

N Bora Keskin and Assaf Zeevi. Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Operations Research*, 62(5):1142–1167, 2014.

Robert Kleinberg and Tom Leighton. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 594–605. IEEE, 2003.

Robert D Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. In *Advances in Neural Information Processing Systems*, pages 697–704, 2005.

Yehuda Koren and Robert Bell. Advances in collaborative filtering. *Recommender systems handbook*, pages 77–118, 2015.

S Rao Kosaraju, Teresa M Przytycka, and Ryan Borgstrom. On an optimal split tree problem. In *Workshop on Algorithms and Data Structures (WADS'99)*, pages 157–168. Springer, 1999.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Yanzhe Murray Lei, Stefanus Jasin, and Amitabh Sinha. Near-optimal bisection search for nonparametric dynamic pricing with inventory constraint. *Ross School of Business Paper*, (1252), 2014.

Nir Levine, Koby Crammer, and Shie Mannor. Rotting bandits. *arXiv preprint arXiv:1702.07274*, 2017.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

Yan Liu and William L Cooper. Optimal dynamic pricing with patient customers. *Operations research*, 63 (6):1307–1319, 2015.

Zhen Liu, Srinivasan Parthasarathy, Anand Ranganathan, and Hao Yang. Near-optimal algorithms for shared filter evaluation in data stream systems. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 133–146, 2008.

Ilan Lobel. Dynamic pricing with heterogeneous patience levels. *Operations Research*, 2020.

D. W. Loveland. Performance bounds for binary testing with arbitrary weights. *Acta Inform.*, 22(1):101–114, 1985.

Michael Luca and Oren Reshef. The effect of price on firm reputation. *Management Science*, 2021.

Domen Malc, Damijan Mumel, and Aleksandra Pisnik. Exploring price fairness perceptions and their influence on consumer behavior. *Journal of Business Research*, 69(9):3693–3697, 2016.

Michael Mitzenmacher and Eli Upfal. *Probability and computing: randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

Mikhail Ju. Moshkov. Greedy algorithm with weights for decision tree construction. *Fundam. Inform.*, 104 (3):285–292, 2010.

Mohammad Naghshvar, Tara Javidi, and Kamalika Chaudhuri. Noisy bayesian active learning. In *50th Annual Allerton Conference on Communication, Control, and Computing, Allerton 2012, Allerton Park & Retreat Center, Monticello, IL, USA, October 1-5, 2012*, pages 1626–1633, 2012.

Feng Nan and Venkatesh Saligrama. Comments on the proof of adaptive stochastic set cover based on adaptive submodularity and its implications for the group identification problem in "group-based active query selection for rapid diagnosis in time-critical situations". *IEEE Trans. Information Theory*, 63 (11):7612–7614, 2017.

Javad Nasiry and Ioana Popescu. Dynamic pricing with loss-averse consumers and peak-end anchoring. *Operations research*, 59(6):1361–1368, 2011.

Fatemeh Navidi, Prabhanjan Kambadur, and Viswanath Nagarajan. Adaptive submodular ranking and routing. *Oper. Res.*, 68(3):856–877, 2020.

Robert D. Nowak. Noisy generalized binary search. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009 (NIPS'09). Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 1366–1374, 2009.

Nishant Oli, Aditya Patel, Vishesh Sharma, Sai Dinesh Dacharaju, and Sushrut Ikhar. Personalizing multimodal content for a diverse audience: A scalable deep learning approach.

Yiangos Papanastasiou and Nicos Savva. Dynamic pricing in the presence of social learning and strategic consumers. *Management Science*, 63(4):919–939, 2017.

Georgia Perakis and Divya Singhvi. Dynamic pricing with unknown non-parametric demand and limited price changes. *Available at SSRN 3336949*, 2019.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, Erik Snowberg, et al. Batched bandit problems. *Annals of Statistics*, 44(2):660–681, 2016.

Greg Petro. Markdown mania: A symptom of the wrong product at the wrong price. In *Total Retail*, page Feb 20, 2017.

Sheng Qiang and Mohsen Bayati. Dynamic pricing with demand covariates. *Available at SSRN 2765257*, 2016.

Rama Ramakrishnan. Markdown management. In *The Oxford Handbook of Pricing Management.* 2012.

Julio J Rotemberg. Customer anger at price increases, time variation in the frequency of price changes and monetary policy. Technical report, National Bureau of Economic Research, 2002.

Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.

Daniel Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. A tutorial on thompson sampling. *arXiv preprint arXiv:1707.02038*, 2017.

Aline Medeiros Saettler, Eduardo Sany Laber, and Ferdinando Cicalese. Trading off worst and expected cost in decision tree problems. *Algorithmica*, 79(3):886–908, 2017.

Jad Salem, Swati Gupta, and Vijay Kamble. Taming wild price fluctuations: Monotone stochastic convex optimization with bandit feedback. *arXiv preprint arXiv:2103.09287*, 2021.

Eric M Schwartz, Eric T Bradlow, and Peter S Fader. Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Science*, 36(4):500–522, 2017.

Aleksandrs Slivkins. Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*, 2019.

Stephen A Smith and Dale D Achabal. Clearance pricing and inventory policies for retail chains. *Management Science*, 44(3):285–300, 1998.

Kalyan Talluri and Garrett Van Ryzin. Revenue management under a general discrete choice model of consumer behavior. *Management Science*, 50(1):15–33, 2004.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Abraham Wald and Jacob Wolfowitz. Optimum character of the sequential probability ratio test. *The Annals of Mathematical Statistics*, 19(3):326–339, 1948.

Yizao Wang, Jean-Yves Audibert, and Rémi Munos. Algorithms for infinitely many-armed bandits. *Advances in Neural Information Processing Systems*, 21, 2008.

Zizhuo Wang. Intertemporal price discrimination via reference price effects. *Operations research*, 64(2):290–296, 2016.

Zizhuo Wang, Shiming Deng, and Yinyu Ye. Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research*, 62(2):318–331, 2014.

Larry Wasserman. *All of nonparametric statistics*. Springer Science & Business Media, 2006.

Laurence A Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.

Shining Wu, Qian Liu, and Rachel Q Zhang. The reference effects on a retailer's dynamic pricing and inventory strategies with strategic consumers. *Operations Research*, 63(6):1320–1335, 2015.

Ya Xu, Nanyu Chen, Addrian Fernandez, Omar Sinno, and Anmol Bhasin. From infrastructure to culture: A/b testing challenges in large scale social networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2227–2236, 2015.

Zikun Ye, Dennis Zhang, Heng Zhang, Renyu Philip Zhang, Xin Chen, and Zhiwei Xu. Cold start to improve market thickness on online advertising platforms: Data-driven algorithms and field experiments. *Available at SSRN 3702786*, 2020.

Rui Yin, Yossi Aviv, Amit Pazgal, and Christopher S Tang. Optimal markdown pricing: Implications of inventory display formats in the presence of strategic customers. *Management Science*, 55(8):1391–1408, 2009.

Jia Yuan Yu and Shie Mannor. Unimodal bandits. In *ICML*, pages 41–48. Citeseer, 2011.