Privacy-Preserving Dynamic Personalized Pricing with Demand Learning

Citation: Chen, Xi, Simchi-Levi, David and Wang, Yining. 2022. "Privacy-Preserving Dynamic Personalized Pricing with Demand Learning." Management Science, 68 (7).

As Published: 10.1287/MNSC.2021.4129

Publisher: Institute for Operations Research and the Management Sciences (INFORMS)

Persistent URL: https://hdl.handle.net/1721.1/148657

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Privacy-Preserving Dynamic Personalized Pricing with Demand Learning

Xi Chen*
Leonard N. Stern School of Business, New York University, xc13@stern.nyu.edu

David Simchi-Levi
Institute for Data, Systems, and Society, Department of Civil and Environmental Engineering and Operations Research Center, Massachusetts Institute of Technology, dslevi@mit.edu

Yining Wang
Warrington College of Business, University of Florida, yining.wang@warrington.ufl.edu

The prevalence of e-commerce has made customers' detailed personal information readily accessible to retailers, and this information has been widely used in pricing decisions. When using personalized information, the question of how to protect the privacy of such information becomes a critical issue in practice. In this paper, we consider a dynamic pricing problem over T time periods with an *unknown* demand function of posted price and personalized information. At each time t, the retailer observes an arriving customer's personal information and offers a price. The customer then makes the purchase decision, which will be utilized by the retailer to learn the underlying demand function. There is potentially a serious privacy concern during this process: a third-party agent might infer the personalized information and purchase decisions from price changes in the pricing system. Using the fundamental framework of differential privacy from computer science, we develop a privacy-preserving dynamic pricing policy, which tries to maximize the retailer revenue while avoiding information leakage of individual customer's information and purchasing decisions. To this end, we first introduce a notion of *anticipating* (ε, δ)-differential privacy that is tailored to the dynamic pricing problem. Our policy achieves both the privacy guarantee and the performance guarantee in terms of regret. Roughly speaking, for d-dimensional personalized information, our algorithm achieves the expected regret at the order of $\widetilde{O}(\varepsilon^{-1}\sqrt{d^3T})$, when the customers' information is adversarially chosen. For stochastic personalized information, the regret bound can be further improved to $\widetilde{O}(\sqrt{d^2T} + \varepsilon^{-2}d^2)$.

Key words: Differential privacy (DP), Dynamic pricing, Generalized linear bandits, Personal information

1. Introduction

The increasing prominence of e-commerce has given retailers an unprecedented power to understand customers as individuals and to tailor their services accordingly. For example, personal information is known to be used in pricing on travel websites (Hannak et al. 2014) and Amazon (Chen et al. 2016); Linden et al. (2003) illustrates how personal information is used in Amazon recommender

* Author names listed in alphabetical order.

systems to achieve a dramatic increase in click-through and conversion rates. Although personalized pricing may involve complicated legal issues in many domains, it has been adopted or considered in several key industries, such as air travel, hotel booking, insurance, and ride-sharing. For example, according to Tringale (2018), "Hotel websites such as Orbitz (whose parent company is Expedia) and auto dealers like Tesla utilize personalized pricing to their advantage when conducting sales with a customer. Even Uber has dabbled in personalized pricing by offering 'premium pricing' to predict which users are willing to pay more to go to a certain location." As reported by Mohammed (2017), when using Orbitz, for identical flights, hotel and type of room, the price of the traveling package found on a laptop was 6.5% more than the price offered on the Orbitz app. Moreover, in practice, instead of directly charging different prices, the e-commerce platforms usually use the discount or promotions to implement personalized pricing strategies.

Although the availability of personal data (e.g., location, web search histories, media consumption, social media activities) enables targeted services for an individual customer, it poses significant privacy issues in practice (e.g., Apple Differential Privacy Team (2017)). Many existing privacy-protection approaches are rather ad hoc by "anonymizing" personal information. However, such ad hoc anonymization leads to two issues. First, it is difficult to quantify the level of privacy. Second, it has been shown that a de-anonymization procedure can easily jeopardize privacy. Examples include the de-anonymization of released AOL search logs (Barbaro & Zeller 2006) and movie-watching records in Netflix challenge (Narayanan & Shmatikov 2008). Therefore, personalized operations management urgently calls for mathematically rigorous privacy-preserving methods to prevent personal information leakage in online decision-making. On one hand, personalized revenue management has received a significant amount of attention in recent operations literature (see, e.g., Ban & Keskin (2021), Cheung & Simchi-Levi (2017) and references therein). On the other hand, the question of how to protect an individual's privacy has not been well-explored in the existing literature.

In this paper, we study how to systematically protect an individual's privacy in the dynamic pricing problem with demand learning. Given T time periods, a potential customer arrives at each time t, and the retailer receives x_t containing information about the incoming customer, such as age, location, purchase history and ratings, credit scores, etc. We consider a very general personalized setting, where the customers are *heterogeneous* and thus the feature $\{x_t\}_{t=1}^T$ does *not* necessarily follow the same distribution. By observing the personal information x_t, the retailer offers the customer a price $p_t \in [0,1]$. The customer then makes $y_t \in \mathbb{R}$, where the random demand y_t follows a *generalized linear model* of a feature vector $\phi(x_t, p_t) \in \mathbb{R}^d$ (see (1)) and the retailer collects revenue $p_t y_t$. The objective of the retailer is to maximize the expected revenue over the entire T time periods, or more specifically $\mathbb{E}[\sum_{t=1}^{T} p_t y_t]$. As this paper focuses on how to protect an individual's sensitive information, we consider a stylized setting of pricing a *single* product, with *unlimited* inventories available.

Due to the personalized nature, the aforementioned pricing procedure involves the use of individuals' sensitive information, such as customers' personal information, characterized by x_t and their purchase history, designated by y_t (e.g., whether a purchase was made at time t). Thanks to secured internet communication channels, the information (x_t, p_t, y_t) at time t is usually securely transmitted, and thus only revealed to the retailer and the particular customer coming at time t. However, although the information at time t is not directly accessible to future customers, the sensitive information is not completely shielded from outside third-party agents (a.k.a. attackers or adversaries) because of the ripple effects of historical customers' data on future pricing decisions in a data-driven pricing system. Indeed, a third-party agent who observes his own posted prices *in the future* can potentially infer an individual's personal information x_t and her purchase decision y_t. We provide two examples showing how the sensitive data at time t could be potentially breached and why such privacy leakage could incur serious challenges to the integrity of the underlying pricing system.

Leakage of purchase activity y_t. For sensitive commodities such as medications, customers' purchasing decisions $\{y_t\}$ must be well protected from the public, as such purchases may potentially reveal purchasers' underlying medical conditions. Some dynamic pricing policies would increase prices facing increased sales volumes for a higher profit. Such behavior might inadvertently leak information about y_t to a third party via the fluctuation of prices. For example, a third-party agent might place orders immediately before and after a person of interest and if he sees a slight spike in his received prices, he might be able to infer the purchase decision y_t of the person of interest.

Leakage of customers' personal information x_t. When making the price decision p_t for an arriving customer at time t, the retailer makes use of the customer's personal information x_t. Some components of x_t, such as the customer's age, credit history, and prior purchases, are highly sensitive and should be protected. Consider a natural pricing policy that is highly "local" to personal information, e.g., posting similar prices to future customers with a similar profile to customer t. A third-party agent could arrive before and after a person of interest with guesses of personal information to detect whether there are noticeable changes in the prices. Then, the agent would be able to infer to some degree about the personal information x_t of the individual of interest.

In summary, it is vital to develop systematic and mathematically rigorous policies that *provably* protect customers' privacy. As we previously discussed, simple data anonymization lacks a theoretical foundation and can be jeopardized. On the other hand, the notion of *differential privacy*

(DP), which was proposed in the computer science field (Dwork et al. 2006a,b), has laid a solid foundation for private data analysis and achieved great success in industries. The DP is not only a gold standard notion in academia but also has been widely adopted by companies, such as Apple (Apple Differential Privacy Team 2017), Google (Erlingsson et al. 2014), Microsoft (Ding et al. 2017), and the U.S. Census Bureau (Abowd 2018). The aim of this paper is therefore to build upon the differential privacy notion to design mathematically rigorous privacy policies with provable utility (regret) guarantees for the dynamic personalized pricing problem.

1.1. Our contributions

The major contributions of this paper can be summarized as follows:

Near-optimal regret of provably privacy-aware pricing policies. Built upon the notion of anticipating differential privacy, we propose a privacy-aware personalized pricing algorithm that enjoys rigorous regret guarantees. More specifically, in a general setting when the personalized information of each coming customer can be adversarially chosen, our policy achieves a regret upper bound of $\widetilde{O}(\varepsilon^{-1}\sqrt{d^3T})$, where ε is the parameter in DP (a smaller ε implies a stronger privacy preservation of the resulting algorithm), d is the dimension of the feature map $\phi(x_t, p_t)$, T is the time horizon, and $\widetilde{O}(\cdot)$ hides logarithmic factors (see Theorem 1). The $\sqrt{T}$ dependency on the time horizon T in this regret upper bound is optimal (Broder & Rusmevichientong 2012).

In addition to the regret upper bound for the general personalized information setting, we also study a "stochastic" setting in which the customer's personal information $\{x_t\}$ is assumed to be stochastic and independently and identically distributed from an unknown non-degenerate distribution. We remark that this is a common assumption/setting studied in the existing literature (Qiang & Bayati 2016, Miao et al. 2019). In this setting, with some changes of hyper-parameters of our proposed algorithm, an improved regret upper bound of $\widetilde{O}(d\sqrt{T}+\varepsilon^{-2}d^2)$ can be proved (see Theorem 2). One attractive property of this bound is that it separates the dependency on conventional problem parameters (i.e., d and T) from privacy-related parameter (i.e., ε). The dominating term (with $T \to \infty$) in this regret bound, namely the $\widetilde{O}(d\sqrt{T})$ term, is *optimal* in both d and T, as shown in (Dani et al. 2008).

In both the general setting and the "stochastic" setting, the regret upper bounds of either $\widetilde{O}(\varepsilon^{-1}\sqrt{d^3T})$ or $\widetilde{O}(d\sqrt{T}+\varepsilon^{-2}d^2)$ also characterize the tradeoffs between customers' privacy protection and the revenue (surplus) of the seller under the designed policy. More specifically, the $\varepsilon > 0$ parameter characterizes the level of customers' privacy protection, with smaller ε corresponding to stronger protection against malicious agents. Clearly, as both regret upper bounds depend inversely on ε, it shows that as the seller seeks stronger protection over the privacy of customers' personalized data, the more he/she will suffer from decreased revenue (and a larger regret). This revenue loss is due to additional efforts/randomization required for data privacy protection.

Finally, the privacy requirements imposed on the seller's policy also have interesting implications on consumer surplus. In Sec. 9.2 of this paper, we provide numerical results to characterize the tradeoffs between consumers' privacy protection and consumer surplus. We find that as the implied privacy protection becomes weaker (i.e., the seller having less ability to discriminate against customers based on their personal data and features, resembling a transition from the first-degree to the third-degree price discrimination), the consumer surplus increases because the seller extracts less of the consumer surplus from his/her limited ability to carry out price discrimination.

Technical contributions. Our proposed framework for privacy-preserving personalized dynamic pricing makes use of several existing privacy-aware learning/releasing techniques, such as the AnalyzeGauss method in online PCAs (Dwork et al. 2014), the tree-based aggregation technique for releasing serial data (Chan et al. 2011), and differentially private empirical risk minimization methods (Kifer et al. 2012, Chaudhuri et al. 2011). On the other hand, the development and analysis of our proposed method make several key technical contributions to the general topic of privacy-aware sequential decision-making in revenue management problems, which we briefly summarize as follows:

1. One salient feature of this paper is the inclusion of customers' personal information x_t as sensitive data that needs to be protected, which is different from existing works (Tang et al. 2020), where only purchase activities y_t are regarded as sensitive data (see Section 2 for more discussions). The objective of protecting privacy in $\{x_t\}$ presents two technical challenges. First, as $\{x_t\}$ and subsequently the feature representations $\{\phi_t\}$ are sensitive data, one cannot directly apply the private follow-the-regularized-leader (FTRL) approach in (Tang et al. 2020) to the dynamic pricing problem. Furthermore, the sensitivity of $\{x_t\}$ implies the sensitivity of $\{p_t\}$ as well, since prices offered to incoming customers must be strongly associated with customers' personal information to achieve good revenue performances. To address these challenges, we build our DP setting on the notion of *anticipating DP* (Shariff & Sheffet 2018), which excludes prices in prior selling periods from the outcome sets of a randomized algorithm.
2. The demand rate function f as a function of price p and personal information x is modeled in this paper as a *generalized linear model* within the exponential family. Despite its apparent similarity to linear models, such generalization results in significant challenges when privacy concerns are considered. In fact, this is still an open problem for generalized linear contextual bandit under the DP guarantee. More specifically, the results of Shariff & Sheffet (2018) on

privacy-aware linear bandits rely heavily on the fact that the ordinary least squares solution is in a closed-form with two simple sufficient statistics: the sample covariance matrix $X^\top X$ and the response-weighted feature vector $X^\top y$. With the post-processing property of DP (which we briefly discuss in Section 4.4), it suffices to obtain privacy-preserved copies of $X^\top X$ and $X^\top y$ at each time. In contrast, parameter estimates in generalized linear models are usually obtained using maximum likelihood estimates (MLE), which do not have simple sufficient statistics. It is nearly impossible to guarantee the privacy and a non-trivial regret simultaneously if the MLE is updated at every period. To overcome this challenge, we make the important observation that the required number of updates of MLEs can be reduced significantly (i.e., only $O(d \log T)$ periods of updates will be sufficient). This key observation allows us to compose differentially private empirical risk minimizers (Kifer et al. 2012) to arrive at a privacy-aware contextual bandit algorithm even without explicit sufficient statistics.

3. The generalized linear model for demand rate modeling resembles existing works on parametric contextual bandits without privacy constraints (Li et al. 2017, Filippi et al. 2010, Wang et al. 2019). One significant limitation of these existing works is that, without assuming stochasticity of the contextual vectors, the optimization of parameter estimates in these works is usually non-convex. Examples include the robustified Z-estimation in (Filippi et al. 2010) and the constrained least-squares formulation in (Wang et al. 2019), both of which are non-convex for some popular generalized linear models such as the logistic regression model. While such non-convexity poses only computational difficulties in non-private bandit algorithms, these challenges become much more significant when privacy constraints are imposed since most existing techniques of DP stochastic optimization require convexity (Kifer et al. 2012, Chaudhuri et al. 2011) and the general privacy-aware non-convex optimization is extremely difficult.

   To overcome this challenge, this paper analyzes a constrained maximum likelihood estimation in a more refined style with a relatively large regularization parameter, demonstrating with high probability that the solution to the constrained MLE lies in the strict interior of the constraint set (see Lemma EC.1 in the supplementary material). This result then implies the first-order KKT condition of the solution, from which the Z-estimation analysis in (Li et al. 2017, Filippi et al. 2010) can be used together with the analysis of an objective-perturbed convex minimization problem to obtain satisfactory regret upper bounds.

1.2. Organization

The rest of the paper is organized as follows. Section 2 discusses the related literature in both dynamic pricing and differential privacy. We set up our pricing models and formalize the anticipating DP in Sections 3 and 4. Our policy is presented in Section 5, which contains two components:

privacy releasers and *price optimizers*. Sections 6 and 7 establish the privacy and regret guarantees, respectively, followed by a conclusion in Section 10. All the technical proofs are relegated to the online supplementary material.

2. Literature Review

This section briefly reviews related research from both the personalized pricing and differential privacy literature.

Personalized dynamic pricing with demand learning. Due to the increasing popularity of online retailing, dynamic pricing with demand learning has become an active research area in revenue management in the past ten years (see, e.g., Araman & Caldentey (2009), Besbes & Zeevi (2009), Farias & Van Roy (2010), Harrison et al. (2012), Broder & Rusmevichientong (2012), den Boer & Zwart (2013), Wang et al. (2014), Chen et al. (2015), Besbes & Zeevi (2015), Cheung et al. (2017), Ferreira et al. (2018), Wang et al. (2021)). More recently, due to the availability of abundant personal information, personalized pricing with feature information has been investigated in several works. For example, Chen et al. (2021) studied offline personalized pricing and quantified the statistical property of the MLE. Cohen et al. (2020) considered a binary thresholding model for purchasing decisions by comparing a linear function of the feature and the posted price, proposed an ellipsoid-based method for dynamic pricing, and established the worst case regret bound. Qiang & Bayati (2016) considered a linear demand model and studied the performance of the greedy iterated least squares. Ban & Keskin (2021) and Javanmard & Nazerzadeh (2019) studied the personalized dynamic pricing problem in high-dimensional settings with sparsity assumption of features. A very recent work by Tang et al. (2020) studied differentially-private contextual dynamic pricing and proposed a Follow-the-Approximate-Leader-type policy. Our work differs from this paper in several aspects. First, we protect the personal information $\{x_t\}$, while Tang et al. (2020) treated this information as public. Second, Tang et al. (2020) adopted the classical DP notion, while we consider the notion of anticipating DP. Finally, we assume that the demand follows a generalized linear model of a feature map of personal information and price, while Tang et al. (2020) considered a binary thresholding purchase model with a linear mapping of contextual information.

Differential privacy for online learning. Since the notation of (ε, δ)-differential (DP) privacy was proposed by Dwork et al. (2006a,b), it has become a golden standard for privacy-preserving data analysis in both academia and industry. Please refer to the survey Dwork & Roth (2014) for a comprehensive introduction of DP.

Built on this classical notion, other privacy notions have also been developed in the literature, such as Gaussian DP (Dong et al. 2019), joint DP (Shariff & Sheffet 2018), local DP (Evfimievski et al. 2003, Kasiviswanathan et al. 2011), average-KL DP (Wang et al. 2016) and per-instance DP (Wang 2019). Our notion of anticipating DP is motivated by the joint DP (Shariff & Sheffet 2018) designed for linear contextual bandits. While the work of Shariff & Sheffet (2018) studied the linear contextual bandits subject to differential privacy constraints, their methods and analysis are built upon the noisy perturbation of sufficient statistics (namely, the sample covariance and sample average). Thus, their method is *not* applicable to the personalized pricing question, where generalized linear demand models are widely used (see also the technical challenges summarized in the introduction).

In DP, there are several fundamental techniques, such as composition, post-processing (see Section 4.3 and Dwork & Roth (2014)), partial-sum by tree-based aggregation Dwork et al. (2010), Chan et al. (2011), and "objective-perturbation" (Chaudhuri et al. 2011, Kifer et al. 2012). In our designed personalized dynamic pricing algorithm, we build on these important techniques to make sure that our algorithm is differentially private.

The techniques of DP have been applied to multi-armed bandit problems. For example, Mishra & Thakurta (2015) developed differentially private UCB and Thompson sampling algorithms for classical bandits. Mishra & Thakurta (2015) and Shariff & Sheffet (2018) further studied differentially private linear contextual bandits, where Mishra & Thakurta (2015) protected the privacy of rewards and Shariff & Sheffet (2018) protected both rewards and contextual information. However, for linear bandits, since the maximum likelihood estimator (MLE) admits a simple closed-form solution, one only needs to protect the sufficient statistics (e.g., $\sum_{t'=1}^{t} x_{t'} x_{t'}^{\top}$ and $\sum_{t'=1}^{t} y_{t'} x_{t'}$). On the other hand, we consider a much more general demand model following a generalized linear model. Therefore, the corresponding MLE does not admit a closed-form solution; we address this challenge by providing a new analysis of constrained MLE properties. There are other interesting private online learning frameworks developed in recent literature. For example, the private sequential learning model was proposed in Tsitsiklis et al. (2020) (for noiseless responses) and further investigated in Xu (2018) and Xu et al. (2020) (for noisy responses). In particular, Xu et al. (2020) quantified the optimal query complexity for private sequential learning against eavesdropping. While existing privacy literature mainly focuses on protecting a data owner's privacy, this work investigates how to protect the privacy of a learner who sequentially queries a database and receives binary responses. We note that the goal of the private sequential learning is to learn a global parameter, e.g., "the highest price to charge so that at least 50% of the consumers would purchase" in pricing domain (Xu et al. 2020), and to make sure the adversary cannot infer the final released price. In contrast, our goal is to make sequential decision-making to maximize revenue while protecting individuals' personalized information and purchasing decisions.

In the recent work of Lei et al. (2020), which was completed after this paper was released, an *offline* personalized pricing setting is studied with differential privacy guarantees. The recent work of Zheng et al. (2020) studied the stronger local privacy notion and derived an algorithm with $\widetilde{O}(T^{3/4})$ regret bound for the generalized linear model, which is worse than the regret bounds obtained in this paper.

3. Pricing models and assumptions

The basic setting of personalized dynamic pricing has been described in the introduction. In this section, we provide more technical details of the problem setting. At each time t with the observed personal information x_t and the posted price p_t, the (random) demand realized by customer at time t is modeled by a Generalized Linear Model (GLM) within the exponential family, taking the form of

$$\Pr[y_t = y | p_t, x_t, \theta^*] = \exp\{\zeta(y\phi_t^\top \theta^* - m(\phi_t^\top \theta^*)) + h(y)\}, \tag{1}$$

where $\phi_t = \phi(x_t, p_t) \in \mathbb{R}^d$ is a known feature map, $\theta^* \in \mathbb{R}^d$ is an unknown linear model, and $\zeta, m(\cdot), h(\cdot)$ are components of the distribution family. Some examples of exponential family distributions include the Gaussian distribution and the Logistic model, which are given at the end of this section. It is easy to verify that $f(\phi_t^\top \theta^*) := m'(\phi_t^\top \theta^*)$ is the expectation of y_t conditioned on p_t, x_t and θ^*. Hence, we can equivalently write Eq. (1) as

$$y_t = f(\phi_t^\top \theta^*) + \xi_t, \tag{2}$$

where $\phi_t = \phi(x_t, p_t)$ and ξ_t are independent random variables satisfying $\mathbb{E}[\xi_t | p_t, x_t] = 0$.

We next specify the filtration process of x_t and p_t. Let $\mathcal{F}_t = \{(x_\tau, y_\tau, p_\tau)\}_{\tau=1}^t$ be the history up to time period t. In the most general setting, the features $\{x_t\}_{t=1}^T$ of the T customers are arbitrarily chosen before the pricing process starts [1]. The price p_t at each time t is subsequently chosen by the dynamic pricing policy conditioned on filtration $\mathcal{F}_{t-1}$ and x_t. The demand y_t is then realized via $y_t = f(\phi_t^\top \theta^*) + \xi_t$, where $\phi_t = \phi(x_t, p_t)$ and $\mathbb{E}[\xi_t | x_t, p_t, \mathcal{F}_{t-1}] = 0$.

Throughout this paper we impose the following conditions on the distribution family, the linear model, and the feature map:

1. There exists a parameter $B_Y < \infty$ such that $|y_t| \leq B_Y$ for all time periods t in all databases D;

[1] This setting is known as the "oblivious adversary" model in the contextual bandit literature. While this model is weaker than the "fully adversarial" one mostly studied in the literature, we adopt the oblivious adversary model for a more convenient treatment of privacy constraints, as $\{x_t\}$ will not depend on the offered prices or the randomly realized demands.

2. Both the feature vectors and the linear model have at most unit norm, or more specifically $\|\phi(x,p)\|_2, \|\theta^*\|_2 \le 1$ for all x,p;
3. The stochastic noises $\{\xi_t\}$ are centered and sub-Gaussian, meaning that $\mathbb{E}[\xi_t|x_t,p_t,\mathcal{F}_{t-1}] = 0$ and there exists $s < \infty$ such that $\mathbb{E}[e^{\lambda\xi_t}|x_t,p_t,\mathcal{F}_{t-1}] \le e^{\lambda^2 s^2/2}$ for all $\lambda \in \mathbb{R}$;
4. $f(\cdot) = m'(\cdot)$ maps $\mathbb{R}$ to $[0,1]$ is continuously differentiable and strictly monotonically increasing. Furthermore, for all $|z| \le 2$, $K^{-1} \le f'(z) \le K$ for some constant $1 \le K < \infty$;
5. ζ in Eq. (1) satisfies $G^{-1} \le \zeta \le G$ for some constant $1 \le G < \infty$.

We give some common examples that fall into Eq. (1) and satisfy all imposed conditions.

Example 1 (Gaussian model). In the Gaussian model the realized demand y_t follows $y_t = \phi_t^\top \theta^* + \xi_t$ with $\xi_t \sim \mathcal{N}(0,1)$. It is easy to verify that the Gaussian model falls into Eq. (2) with $\zeta = 1$, $m(z) = \frac{1}{2}z^2$, $f(z) = m'(z) = z$, and $h(y) = -\frac{1}{2}y^2 - \frac{1}{2}\ln(2\pi)$. The Gaussian model also satisfies all imposed conditions with high probability with $B_Y \lesssim s\sqrt{\ln T}$, $s = 1$, $K = 1$, and $G = 1$.

Example 2 (Logistic model). In the Logistic model the realized demand y_t is supported on $\{0,1\}$, following the Logistic distribution $\Pr[y_t = 1|\phi_t,\theta^*] = e^{\phi_t^\top\theta^*}/(1+e^{\phi_t^\top\theta^*})$. It is easy to verify that the Logistic model falls into Eq. (2) with $\zeta = 1$, $m(z) = \ln(1+e^z)$, $f(z) = m'(z) = e^z/(1+e^z)$, and $h(y) = 1$. The Logistic model also satisfies all imposed conditions with $B_Y = 1$, $s = 1$, $K = (1+e^2)^2/e^2$, and $G = 1$.

4. Preliminaries on differential privacy

In this section we present background material on *differential privacy*, the core privacy concept adopted in this paper. We start with the introduction of the standard differential privacy concept, and then show how the privacy concept could be extended to its "anticipating" version which is more appropriate for data-driven sequential decision-making problems. Finally we discuss two fundamental concepts of *composition* and *post-processing*, which are essential in designing complex differentially private systems. For a full technical treatment and historical motivations, the readers are referred to the comprehensive review by Dwork & Roth (2014).

4.1. Differential privacy

Differential privacy is a mathematically rigorous measure of privacy protection and has been extensively studied and applied since its proposal in the work of Dwork et al. (2006b). At a higher level, the fundamental concept behind differential privacy is the *impossibility* of distinguishing two "neighboring databases" (differing only on a single entry) with high probability, based on publicly available information about the database. To facilitate such probabilistic indistinguishability, the

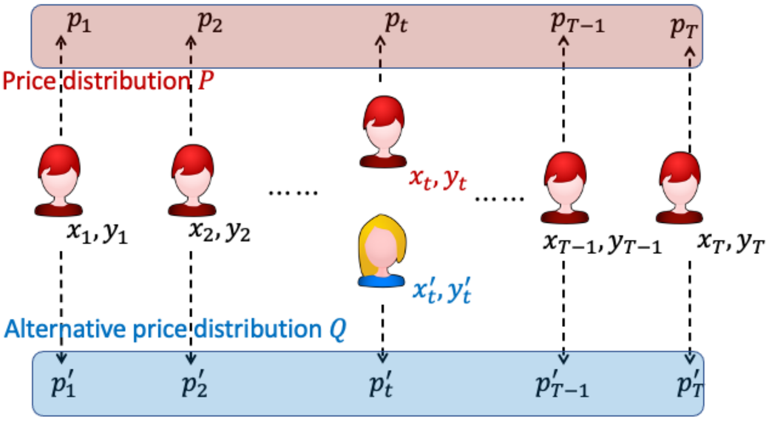

Figure 1 **Illustration of the differential privacy concept.**

conventional approach is to artificially calibrate *stochastic noise* into the process or the outputs of differentially private algorithms.

More specifically, Figure 1 gives an intuitive illustration of the differential privacy concept applied to our dynamic personalized pricing problem. Suppose at time t the incoming customer with the context vector x_t is being offered price p_t and makes purchase decision y_t. The price decisions $\{p_t\}_{t=1}^T$ produced by the pricing algorithm are usually random, and therefore we can use P to denote the joint distribution of these prices. The concept of differential privacy requires that, if a customer's personal data change from (x_t, y_t) to (x'_t, y'_t), while all the other $T-1$ customers' data remain unchanged, the joint distributions of the posted prices P will change to a distribution Q that is very close to P. The closer P and Q are under the hypothetical personal data change $(x_t, y_t) \to (x'_t, y'_t)$, the better data privacy is protected under the pricing policy.

Why is the close proximity of price distributions P and Q a good measurement of a pricing algorithm's privacy protection? Assume that a malicious agent would like to extract the sensitive information of a particular customer of interest, who arrives in the system at time t. The malicious agent must extract such sensitive data based solely on *publicly available* information, which in this case would be the firm's posted prices $p_1, \cdots, p_T$. Here, "public information" in the differential privacy literature refers to the information or released data that can be accessed by a malicious adversary, because these data are used by the adversary to infer the personalized data of the customers, whose privacy is to be protected. If the price distributions P and Q produced by the pricing algorithm are very similar, then it is *information-theoretically* not possible for the malicious agent to distinguish with reasonable success probability between a customer (x_t, y_t) and another hypothetical customer (x'_t, y'_t) (see Figure 1). This means that no matter how smart the malicious agent is, it is impossible for him to extract very much sensitive data from the customer of interest simply based on publicly available price information.

Mathematically, we use D to denote the database of all sensitive data $\{(x_t, y_t)\}_{t=1}^T$ for all of the T customers. For convenience of presentation, we also write $o_t = (x_t, y_t)$. A database D' that is a *neighboring database* of D if and only if D' and D only differ at a single time period. More specifically, $D = \{o_t\}_{t=1}^T, D' = \{o'_t\}_{t=1}^T$ are neighboring databases if there exists t such that $o_t \neq o'_t$ and $o_\tau = o'_\tau$ for all $\tau \neq t$. Suppose a pricing algorithm A operates with input database D and produces randomized price output $A(D) = (p_1, \cdots, p_T)$. The following definition gives a rigorous formulation of (ε, δ)-differential privacy:

Definition 1 ((ε, δ)-differential privacy (Dwork et al. 2006a)). For $\varepsilon, \delta > 0$, a randomized algorithm A satisfies (ε, δ)-differential privacy if for every pair of neighboring databases D, D' and measurable set $\mathcal{A} \subseteq [\underline{p}, \overline{p}]^T$, it holds that

$$\Pr[A(D) \in \mathcal{A}] \leq e^\varepsilon \Pr[A(D') \in \mathcal{A}] + \delta.$$

To facilitate the understanding of this definition, we explain why the multiplicative factor e^ε is critical and the role of the parameter δ in practice. Let us first explain why the DP-definition in Def. 1 adopts a multiplicative factor e^ε rather than an additive bound of $|\Pr[A(D) \in \mathcal{A}] - \Pr[A(D') \in \mathcal{A}]$. Imagine two neighboring datasets D, D' give rise to the same output O with probabilities $p_1 = \Pr[O|D]$ and $p_2 = \Pr[O|D']$. The key is to prevent a malicious party from distinguishing between D and D' based on the observation of O. If an additive guarantee is involved $|p_1 - p_2| \leq \varepsilon$, then it is possible that $p_1 = 0$ and $p_2 = \varepsilon$. If this is the case, the adversary would be *100% sure whether the underlying dataset is D or D'* once she observes the output O (since $p_1 = 0$ implies that it is *impossible* to observe O given D). This means that with probability ε, which is usually not that small (e.g., $\varepsilon = 0.1$), a *catastrophe* (i.e., an outside adversary being *completely certain* about the customer's private data) will occur with 10% probability. On the other hand, if the guarantee is multiplicative (e.g., $0.9p_2 \leq p_1 \leq 1.1p_2$) then the adversary *cannot* completely distinguish between D and D' no matter how small p_1 or p_2 is. Following this discussion on the multiplicative factor versus the additive factor, since δ is an additive term, it corresponds to the probability of a *catastrophe* happening that allows the adversary to completely infer the privacy information about customers' data. Since we don't want a catastrophe to happen, δ needs to be set *overwhelmingly* small. With a tiny δ value in the DP-definition, more specifically, the adversary is *always* able to conclude that D (or D') is more likely than the other, but such preference of likelihood is never going to exceed a ratio of e^ε. For example, with $\varepsilon = 0.1$, the adversary may conclude that D is

10.5% more possible than D' based on his observations of published data O, but will never be able to completely/deterministically distinguish D from D' based on O.

4.2. Anticipating differential privacy

Despite being a widely adopted measure, the DP notion as stated in Definition 1 cannot be directly applied to dynamic pricing for several reasons. First, Definition 1 would not lead to useful pricing policies. This is because, essentially, Definition 1 requires that conditioned on the output of the *entire* posted price sequence, the adversary cannot distinguish between o_t and o'_t in a probabilistic sense. On the other hand, for high-profit personalized pricing policies, once the customer's personal information x_t changes, the price p_t offered to that customer must change accordingly in order to achieve high expected revenue, making inference of x_t much easier given p_t. Furthermore, as we have discussed in the previous paragraphs, the communications of (x_t, p_t, y_t) at time t are secured in practice and therefore, an adversary should not have the capability of accessing the price p_t at time t. From this perspective, the classical DP notion defined in Definition 1 is too strong since it implicitly allows the adversary to access the price at time t (as p_t belongs to the output $A(D)$). In a practical setting, however, the adversary is only able to access information during other time periods (e.g., by maliciously sending fake customers to obtain price quotes) to infer the sensitive information about an individual at time t. In other words, in the following anticipating DP definition (see Definition 2), the price offered to a specific customer of interest p_t is not public information, as we can expect basic communication security between the customer and the seller. However, prices offered to *other* customers are considered public information because a malicious adversary could pretend to be a customer and extract such price information, and subsequently infer the private data of the customer of interest based on such extracted price information.

This argument can be made rigorous by the following proposition. The proposition is similar to Claim 13 of Shariff & Sheffet (2018), by showing that *any* policy satisfying the (ε, δ)-differential privacy in Definition 1 must suffer regret that is linear in the time horizon T. The proof of Proposition 1 is, however, different from Shariff & Sheffet (2018), since we study generalized linear models such as the logistic regression model. We relegate the complete proof to the supplementary material.

PROPOSITION 1. *Let π be a contextual pricing policy over T periods that satisfies (ε, δ)-differential privacy as defined in Definition 1, with $\varepsilon < \ln(2)$ and $\delta < 1/4$. Then the worst case regret of π is lower bounded by $\Omega(T)$.*

To address the challenges mentioned, Shariff & Sheffet (2018) proposed a notion of "joint DP" in the context of linear contextual bandits. We adopt this notion but refer to it as *anticipating DP*.

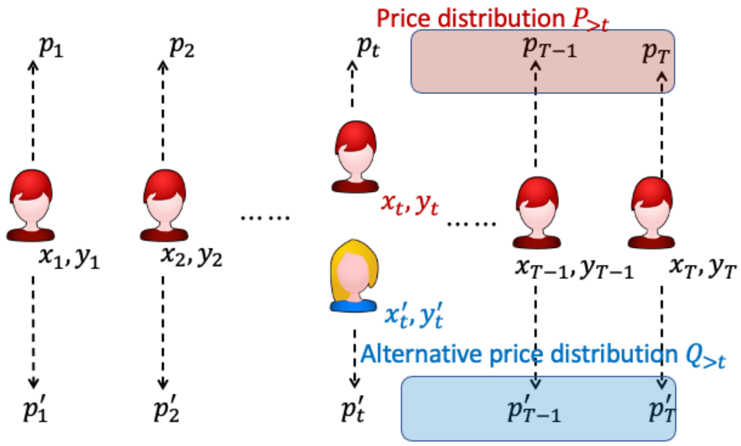

Figure 2 **Illustration of the anticipating differential privacy (ADP) concept.**

The notion of anticipating DP highlights the key property of this definition and our focus on more general dynamic personalized pricing policies. Figure 2 gives an illustration of the anticipating differential privacy (ADP) concept. Compared to the classical differential privacy notion illustrated in Figure 1, the important difference of ADP is to restrict the output sets to prices strictly *after* a customer of interest t and to only require the distributions of *anticipating* prices (denoted by $P_{>t}$ and $Q_{>t}$) to remain stable with change of personal information $(x_t, y_t) \to (x_t', y_t')$ at time t. Such a restriction is motivated by the fact that the communication about (x_t, p_t, y_t) at time t is secured and the data prior to time t has no impact on the privacy of customer t since the pricing algorithm has no knowledge of x_t before time t. With the formulation of anticipating differential privacy, the challenges we mentioned earlier are resolved because the pricing decision p_t at time t is no longer in the information set of a potential attacker.

Our next definition gives a rigorous mathematical formulation of the anticipating differential privacy notion illustrated in Figure 2.

Definition 2 (anticipating (ε,δ)-differential privacy). Let $\varepsilon, \delta > 0$ be privacy parameters. A dynamic personalized pricing policy π satisfies anticipating (ε,δ)-differential privacy if for any pair of neighboring databases D, D' differing at time t (i.e., $o_t \neq o_t'$) and measurable set $\mathcal{P}_{>t}$, it holds that

$$\Pr[p_{t+1}, \cdots, p_T \in \mathcal{P}_{>t} | \pi, D] \le e^{\varepsilon} \Pr[p_{t+1}, \cdots, p_T \in \mathcal{P}_{>t} | \pi, D'] + \delta. \tag{3}$$

We also remark that all privacy definitions in this section are *model-free*, meaning that they do *not* depend on how realized demands y_t are modeled. Hence, the privacy guarantees of our proposed algorithm are independent from the generalized linear demand model in Eqs. (1, 2). This fact is essential in practical implementations of privacy-aware algorithms because one cannot build privacy guarantees of an algorithm on a specific underlying model, which may or may not hold in reality. The modeling assumptions, on the other hand, are required for *performance analysis* (also known as *utility analysis*, e.g., regret upper bounds or convergence results) of our proposed privacy-aware pricing policies.

4.3. Composition in differential privacy

When a differentially private algorithm only outputs a single statistic (e.g., the sample mean of the database), Definition 1 is easy to check and verify. In reality, however, a useful differentially private protocol is tasked to release several statistics (sometimes with adaptively chosen queries) and the *entire* output sequence of a protocol needs to be differentially private. With multiple output statistics, Definition 1 involves high-dimensional vector spaces and is therefore difficult to check and verify. *Composition*, on the other hand, provides convenient *upper bounds* on the privacy guarantee of composite outputs using privacy guarantees of individual queries. Take the dynamic pricing setting as an example. The seller repeatedly interacts with the potential customers by offering different prices. It is therefore essential to leverage a composition guarantee in Fact 1 to make sure that all the prices offered, *when aggregated as a whole*, do not leak consumers' privacy via their personalized data.

The left panel of Figure 3 gives an illustration of the concept of composition in the context of personalized pricing. In this simple example, a centralized pricing algorithm has access to a pool of past customers' sensitive data and offers personalized prices to three customers. The rule of composition in differential privacy asserts that the privacy guarantee of the pricing algorithm *worsens* as the pricing algorithm offers prices to more customers, each time with access and calculations based on the majority of the same sensitive data. In particular, if the privacy guarantee for each individual pricing decision is ε, then the joint privacy guarantee when k individualized prices are offered will worsen to $\Omega(k\varepsilon)$ or $\Omega(\sqrt{k}\varepsilon)$, depending on the detailed composition mechanisms.

More specifically, let $A = (A_1, \cdots, A_k)$ be a collection of k adaptively chosen queries and suppose that each query A_k satisfies (ε, δ)-differential privacy as defined in Definition 1. The following result is standard in the literature and cited from Theorems 3.16 and 3.20 from (Dwork & Roth 2014).

Fact 1 *The composite query $A = (A_1, \cdots, A_k)$ satisfies (ε', δ')-differential privacy with either one of the following:*

1. (Basic composition) $\varepsilon' = k\varepsilon$, $\delta' = k\delta$;
2. (Advanced composition) $\varepsilon' = \sqrt{2k\ln(1/\widetilde{\delta})}\varepsilon + k\varepsilon(e^{\varepsilon} - 1)$, $\delta' = k\delta + \widetilde{\delta}$ *for* $\widetilde{\delta} > 0$.

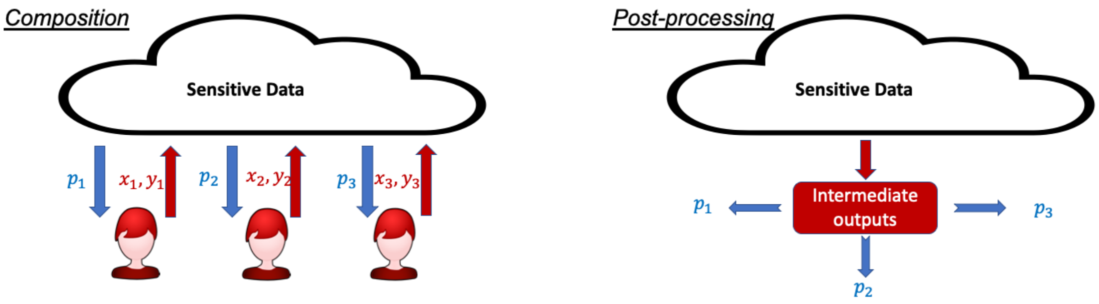

Figure 3 **Illustration of the concepts of composition (left) and post-processing (right) in differential privacy.**

To avoid potential confusion, we remark that both basic and advanced composition apply to *any* differentially private algorithms. Indeed, they are two different types of joint privacy guarantees proved using different techniques, reflecting different tradeoffs when composing multiple differentially private queries/algorithms together. In particular, the basic composition shows a linear growth in the ε parameter (i.e., $\varepsilon' = k\varepsilon$) but it allows the δ' parameter to be zero when the individual queries are $(\varepsilon, 0)$-private. On the other hand, the advanced composition allows for a slower growth of the ε parameter (i.e., $\varepsilon' \asymp \sqrt{k}\varepsilon$) but must yield an (ε', δ') differential privacy guarantee with $\delta' > 0$, even if the individual queries are $(\varepsilon, 0)$-private. In this paper, we shall use primarily the advanced composition result because we focus on (ε, δ) privacy guarantees with $\delta > 0$.

Corollary 1 **(Corollary 3.21, (Dwork & Roth 2014))**. *Given target privacy level* $0 < \varepsilon' < 1$, $\delta' > 0$ *of the composite query* A*, it is sufficient for each sub-query to be* (ε, δ)*-differentially private with* $\varepsilon = \varepsilon'/2\sqrt{2k\ln(2k/\delta)}$ *and* $\delta = \delta'/2k$.

4.4. Post-processing in differential privacy

Practical privacy-aware algorithms usually involve several separate sub-routines. In most of the cases, not all sub-routines access the sensitive database: some sub-routines may only process the results from other sub-routines. The principle of *post-processing* states that one only needs to preserve the privacy of those sub-routines with access to the sensitive database in order to argue for privacy protection of the entire algorithm. For example, in dynamic pricing, algorithms are developed into different components and only one of them directly accesses the sensitive data. It is therefore necessary to use the concept of post-processing to argue that the entire algorithm viewed as a whole does not leak consumers' private personalized data.

The right panel of Figure 3 gives an intuitive illustration of the post-processing concept in differential privacy. Suppose that an algorithm with full access to all sensitive data has produced some intermediate results (as shown in the red square of the illustration), and these intermediate results have already satisfied the definitions of differential privacy. Further assume that there is a downstream algorithm, which operates arbitrarily on the intermediate results to produce the personalized prices $p_1, p_2, p_3, \ldots$, *without accessing the sensitive data any more.* Then the *post-processing* asserts that there is no need to worry about potential privacy leakages of the downstream algorithm because the intermediate results have already been privatized. This useful concept makes it easier to design multi-step, sophisticated privacy-preserving algorithms.

More specifically, let A be a sub-routine with access to the sensitive database and B be a sub-routine that only depends on the results of A.

Fact 2 (Proposition 2.1, (Dwork & Roth 2014)) *Suppose the outputs of sub-routine A satisfy (ε, δ)-differential privacy. Then the outputs of sub-routine B also satisfy (ε, δ)-differential privacy.*

5. Algorithmic framework

In this section we present the framework of our proposed privacy-aware dynamic personalized pricing algorithm.

A straightforward idea is to directly inject noise into customers' sensitive information (e.g., x_t) to protect privacy. However, as we will explain later in the paper (see Section 9.1), such a method will fail because the features of each individual customer are relatively independent of each other. Thus, an excessively large magnitude of noise needs to be injected, which incurs a large regret. Therefore, this paper will develop a new dynamic personalized pricing algorithm based on the privacy-preserving maximum likelihood estimator. To better illustrate our algorithm, we first introduce two types of routines used in our algorithm: the *private releasers* that access the sensitive database and produce differentially private outputs, and the *price optimizers* that access only the outputs from private releasers to assign near-optimal and privacy-aware prices. Then a pseudo-code description of our main algorithm will be presented and discussed.

5.1. Private releasers and price optimizers

Our proposed privacy-preserving dynamic personalized pricing algorithm consists of several sub-routines. We divide the sub-routines into two classes: the *private releasers* and the *price optimizers*.

The *private releasers* access the sensitive database $\{x_t, p_t, y_t\}_{t=1}^{T}$ and output differentially private intermediate results. For example, in Figure 4 the PRIVATECOV routine returns differentially private sample covariance matrices and the PRIVATEMLE routine returns differentially private maximum likelihood estimates. For private releaser routines, the differential privacy notions are classical (in Definition 1). Note that, in addition to differential privacy guarantees, the sub-routines also need to satisfy the anticipating constraints for pricing algorithms (i.e., accessing only $\{x_\tau, y_\tau, p_\tau\}_{\tau<t}$ to produce any outputs being used at time t).

The *price optimizer*, on the other hand, performs optimization and outputs the prices p_t for each time period t. To ensure privacy, our designed price optimizer will *not* directly access historical sensitive data $\{x_\tau, y_\tau, p_\tau\}_{\tau<t}$. Instead, it optimizes the offering price p_t based only on x_t (the personal information of the incoming customer) and intermediate quantities computed by private releasers up to time t.

Because our designed price optimizer has access to x_t at time t, one cannot directly apply the post-processing rule in Fact 2 to argue privacy guarantees. Nevertheless, the following proposition shows that if all private releasers are differentially private, then so is the price optimizer in the sense of anticipating differential privacy in Definition 2. The proof of Proposition 2 is placed in the supplementary material.

Proposition 2. *Let $(a_1, \cdots, a_T)$ be the outputs of private releasers at each time period t and suppose the entire output sequence $(a_1, \cdots, a_T)$ satisfies (ε, δ)-differential privacy. Suppose the price p_t at time t is a deterministic function of x_t and $a_1, \cdots, a_{t-1}$. Then the pricing policy satisfies anticipating (ε, δ)-differential privacy.*

Remark 1. The conclusion in Proposition 2 holds for p_t as randomized functions of x_t, $a_1, \cdots, a_{t-1}$ as well. Nevertheless, because in our proposed algorithm the price optimizer is deterministic, we shall restrict ourselves to deterministic functions.

5.2. Our policy

In Figure 4 we depict a high-level framework of our privacy-aware dynamic personalized pricing policy. It shows a three-layer structure of the proposed policy. The first layer is the *sensitive database*, consisting of data $\{o_t = (p_t, x_t, y_t)\}_{t=1}^T$; and its privacy needs to be protected. The second layer is *private releasers*, which consists of two sub-routines PrivateCov (see Algorithm 2 in Section 6.1) and PrivateMLE (see Algorithm 3 in Section 6.2). The PrivateCov sub-routine supplies differentially private sample covariance matrices $\Lambda_n^p \in \mathbb{R}^{d\times d}$ at every time period. The PrivateMLE sub-routine outputs differentially private maximum likelihood estimates $\widehat{\theta}_n^p$, but only when such estimates are requested by the price optimizer. The PrivateCov sub-routine is designed to be $(\varepsilon_1, \delta_1)$-differentially private and the PrivateMLE routine is $(\varepsilon_2, \delta_2)$-differentially private, so that all outputs from private releasers are $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differentially private, thanks to the basic composition rule in Fact 1.

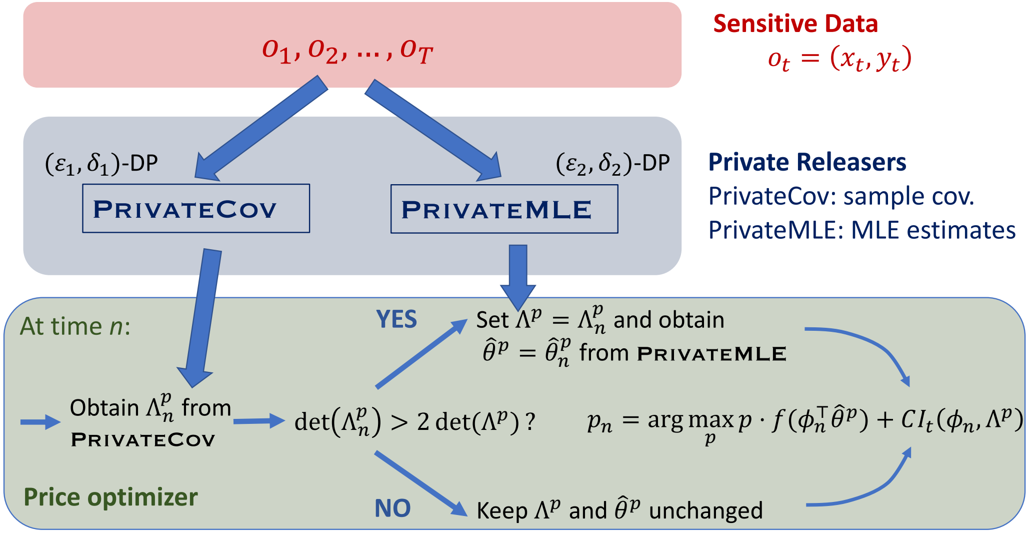
$o_1, o_2, \dots, o_T$
Sensitive Data
$o_t = (x_t, y_t)$
$(\varepsilon_1, \delta_1)$-DP
$(\varepsilon_2, \delta_2)$-DP
PRIVATECOV
PRIVATEMLE
Private Releasers
PrivateCov: sample cov.
PrivateMLE: MLE estimates
At time n:
Obtain Λ_n^p from PRIVATECOV
$\det(\Lambda_n^p) > 2\det(\Lambda^p)$?
YES
Set $\Lambda^p = \Lambda_n^p$ and obtain $\hat\theta^p = \hat\theta_n^p$ from PRIVATEMLE
NO
Keep Λ^p and $\hat\theta^p$ unchanged
$p_n = \arg\max_p p \cdot f(\phi_n^\top \hat\theta^p) + CI_t(\phi_n, \Lambda^p)$
Price optimizer

Figure 4 Our algorithm framework. Details and explanations in Section 5.2 in the main text.

The third layer of our proposed policy is the price optimizer. As discussed in the previous section, to ensure privacy the price optimizer shall not access the sensitive database D directly. Instead it should base its decision of p_t on outputs from private releasers and x_t only. The last block in Figure 4 illustrates the basic flow of our price optimizer. The price optimizer maintains Λ^p and $\widehat{\theta}^p$ throughout the pricing process, both of which are obtained directly from private releasers without accessing the sensitive database. At the beginning of time period n, the price optimizer first obtains sample covariance Λ_n^p from the PRIVATECOV routine. The optimizer then decides whether to request fresh MLE from the PRIVATEMLE routine by comparing $\det(\Lambda_n^p)$ with $\det(\Lambda^p)$, in addition to some other criteria specified in Algorithm 1. Afterwards, p_t is selected as the maximizer of an upper confidence bound of the expected revenue on x_t. It is only during this step that the personal information x_t is involved.

Algorithm 1 also gives a pseudo-code description of our proposed pricing policy, which is more accurate and detailed than Figure 4. Note that Algorithm 1 involves several algorithmic parameters, such as T_0, D_∞, γ, and ρ, which do *not* affect the privacy guarantees of the algorithm but do have an impact on its performance. How to set these algorithmic parameters will be given later in Section 7 when we analyze the regret performance of Algorithm 1. Before that, we will first make a few important remarks about Algorithm 1.

REMARK 2 (TIME COMPLEXITY). The time complexity for the PRIVATEMLE sub-routine is the same as traditional maximum likelihood estimation calculations, if not easier (since the overall formulation is convex), because only the objective is perturbed with a linear term. The time

Algorithm 1 The framework of privacy-aware dynamic personalized pricing

```
Input: privacy parameters $\varepsilon_1, \delta_1, \varepsilon_2, \delta_2 > 0$, number of pure-exploration periods $T_0$, maximum number of PrivateMLE calls $D_\infty$, regularization parameter $\rho \geq 1$, confidence parameter $\gamma > 0$.
Output: the offering prices $p_1, p_2, \cdots, p_T$;
$\delta_2' = \frac{\delta_2}{2D_\infty}$, $\varepsilon_2' \leftarrow \frac{\varepsilon_2}{2\sqrt{2D_\infty \ln(1/\delta_2')}}$, $\Lambda^p = \rho I_d$, $\widehat{\theta}^p = 0$, $D_{\text{MLE}} = 0$;
For the first $T_0$ time periods, offer prices $p_t$ uniformly at random from $[0,1]$;
for $n = T_0 + 1, \cdots, T$ do
    Obtain $\Sigma_n^p \leftarrow$ PrivateCov$(n, \varepsilon_1, \delta_1)$ and let $\Lambda_n^p = \Sigma_n^p + \rho I_d$;
    if $\det(\Lambda_n^p) > 2\det(\Lambda^p)$ and $D_{\text{MLE}} < D_\infty$ then
        $\widehat{\theta}^p \leftarrow$ PrivateMLE$(n, \rho, \varepsilon_2', \delta_2')$, $\Lambda^p \leftarrow \Lambda_n^p$, $D_{\text{MLE}} \leftarrow D_{\text{MLE}} + 1$;
    end if
    Offer price $p_n = \arg\max_{p \in [0,1]} \min\{1, pf(\phi_n^\top \widehat{\theta}^p) + \gamma\sqrt{\phi_n^\top (\Lambda^p)^{-1}\phi_n}\}$, where $\phi_n = \phi(x_n, p_n)$;
end for
```

complexity for the PrivateCov sub-routine is slightly more expensive: at each time n, the tree-based protocol needs to update $O(\log n)$ nodes on the binary tree instead of just adding $\phi_t\phi_t^\top$ to a counting matrix. Overall, the algorithm's time complexity is $O(d^3 T \ln T)$ (note d^3 comes from the computation of the determinant), in addition to $O(d \ln T)$ number of MLE calculations. The next section gives more details on the two private releasers.

Remark 3 (Difference from Generalized Linear Contextual Bandit). This remark explains how the algorithm differs from a classic generalized linear bandit algorithm without privacy consideration. The major difference is that when there is no privacy consideration, there is no need (and no use) to randomize and therefore vanilla maximum likelihood estimation (MLE) can be used to obtain an estimated model $\widehat{\theta}_t$ at every time period t, with standard statistical analysis of the errors for such estimates (see Li et al. (2017)). With privacy constraints, such maximum likelihood estimates need to be carefully privatized by calibrating artificial noise into the objective of the MLE (the PrivateMLE sub-routine later in Algorithm 3), which also calls for more detailed perturbation-based statistical analysis. Another difference is that without privacy constraints, the seller could update its model estimate $\widehat{\theta}_t$ at *every* time period to obtain the most accurate and updated information. With privacy constraints, however, the seller cannot afford to adaptively compute a model estimate after each time period due to composition constraints and must perform such model estimates sparingly, relying further on a signal scheme also privatized by incorporating artificial noise matrices (see the PrivateCov sub-routine later in Algorithm 2).

Remark 4 (Exploration Phase). In addition, we also clarify that the forced exploration step in our algorithm is optional: the proposed algorithm remains valid (i.e., satisfying suitable

Algorithm 2 The PrivateCov sub-routine

```
function PrivateCov($T, \varepsilon, \delta$)                                  ▷ returns $\Sigma_1^p, \cdots, \Sigma_{T-1}^p$
    $\delta' \leftarrow \frac{\delta}{2\lceil\log_2 T\rceil}$, $\varepsilon' \leftarrow \frac{\varepsilon}{2\lceil\log_2 T\rceil \ln(1/\delta')}$, $\sigma^2_{\varepsilon',\delta'} = \frac{2\ln(1.25/\delta')}{(\varepsilon')^2}$, $m = \lceil\log_2 T\rceil$;
    Initialize $\Sigma(\ell) = \widehat{\Sigma}(\ell) = 0$ for all $\ell = 0, \cdots, m-1$;
    for $n = 1, 2, \cdots, T-1$ do
        Express $n$ in its binary form: $n = \sum_{\ell=0}^{m-1} b_n(\ell) 2^\ell$, $b_n(\ell) \in \{0, 1\}$;
        Let $\ell_n \leftarrow \min\{\ell : b_n(\ell) = 1\}$ be the least significant bit of $n$;
        Update $\Sigma(\ell_n) \leftarrow \phi_n\phi_n^\top + \sum_{\ell<\ell_n} \Sigma(\ell)$ and $\Sigma(\ell) \leftarrow \widehat{\Sigma}(\ell) \leftarrow 0$ for all $\ell < \ell_n$;
        Calibrate noise: $\widehat{\Sigma}(\ell_n) \leftarrow \Sigma(\ell_n) + W^n$ where $W^n_{ij} = W^n_{ji} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{\varepsilon',\delta'})$;
        Release $\Sigma_n^p = \sum_{\ell=0}^{m-1} b_n(\ell)\widehat{\Sigma}(\ell)$;
    end for
end function
```

differential privacy constraints and achieving small overall regret) without the forced exploration step (see Theorem 1 in Sec. 7.1 where $T_0 = 0$). The forced exploration helps to ensure *improved* regret guarantee when there are additional distributional assumptions on contextual vectors (see Section 7.2). This forced exploration aims to make sure the sample covariance of the context vectors is well-conditioned, which leads to improved regret guarantees of privatized MLE.

6. Design and analysis of private releasers

In this section, we give detailed designs of the two private releasers: the PrivateCov sub-routine and the PrivateMLE sub-routine. We prove that both of them satisfy (ε, δ)-differential privacy as defined in Definition 1. We also prove several utility guarantees that will be helpful later in the regret analysis of the pricing policy. Figure 5 shows the flow of our proof framework. Due to space constraints and exposition concerns, all proofs to technical lemmas or propositions in this section are placed in the supplementary material.

6.1. The Priva teCov sub-routine

Algorithm 2 gives a pseudo-code description of the PrivateCov sub-routine. Note that in Algorithm 2 the Σ_n^p covariance matrices are released sequentially once each time period, and PrivateCov(n, ε, δ) would simply be the Σ_n^p matrix released at the end of iteration $n-1$.

Algorithm 2 is based on the AnalyzeGauss framework in (Dwork et al. 2014) coupled with the *tree-based aggregation* technique for releasing continual observations (Dwork et al. 2010, Chan et al.

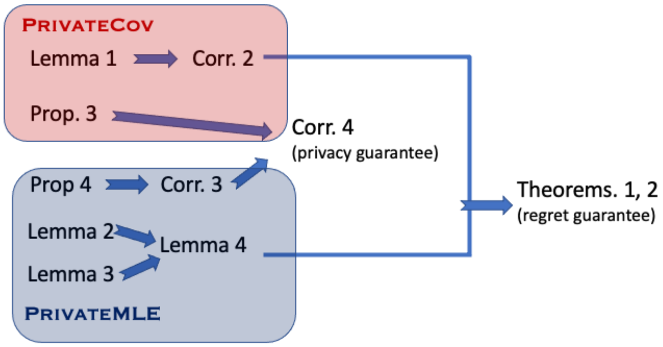

Figure 5 **Flow of our proof framework.**

2011). The AnalyzeGauss by Dwork et al. (2014) develops a Gaussian mechanism on releasing a single covariance matrix privately from the data. On the other hand, tree-based aggregation provides a general protocol on how to continually release sequentially updated statistics (e.g., partial sums of sample covariance matrices) under privacy constraints. For our PrivateCov, by calibrating symmetric random Gaussian matrices $\{W^n\}$ into the sample covariances under the tree-based aggregation, one achieves differential privacy. The following proposition claims that the outputs $(\Sigma_1^p, \cdots, \Sigma_{T-1}^p)$ of Algorithm 2 satisfy (ε, δ)-differential privacy.

Proposition 3. *The outputs of Algorithm 2, $(\Sigma_1^p, \ldots, \Sigma_{T-1}^p)$ satisfy (ε, δ)-differential privacy.*

The following lemma further gives high probability bounds on the deviation from Σ_n^p to the actual sample covariance $\Sigma_n = \sum_{t=1}^n \phi_t \phi_t^\top$. This utility guarantee is useful later in the regret analysis to justify the $\det(\Lambda_n^p) > 2\det(\Lambda^p)$ condition in Algorithm 1.

Lemma 1. *With probability $1 - O(T^{-1})$, it holds for all $n \in \{1, 2, \cdots, T-1\}$ that*

$$\|\Sigma_n^p - \Sigma_n\|_{\mathrm{op}} \le O(\varepsilon^{-1}\sqrt{d}\ln^{4.5}(T/\delta)),$$

where $\Sigma_n = \sum_{t \le n} \phi_t \phi_t^\top$.

The following corollary is an immediate consequence of Lemma 1.

Corollary 2. *Let $\Lambda_n = \Sigma_n + \rho I_d$ and $\Lambda_n^p = \Sigma_n^p + \rho I_d$ for some $\rho \ge \varepsilon^{-1} d\sqrt{d}\ln^5(T/\delta)$. Then there exists a universal constant $C_T < \infty$ such that, for any $T \ge C_T$, with probability $1 - O(T^{-1})$ for all $n \in \{1, 2, \cdots, T-1\}$, it holds that $0.9\det(\Lambda_n) \le \det(\Lambda_n^p) \le 1.11\det(\Lambda_n)$.*

Corollary 2 shows that when the PrivateMLE is invoked in Algorithm 1, the determinant of the real sample covariance matrix roughly doubles. This is important to our later regret analysis because if PrivateMLE is invoked too frequently, the algorithm pays the price of composition in privacy. While on the other hand, if PrivateMLE is invoked too rarely, the old (and inaccurate) parameters will be used for a long time, which incurs a larger regret. Therefore, our analysis shows that the right frequency should be invoking PrivateMLE once the determinant of the privacy-preserving covariance roughly doubles.

6.2. The PrivateMLE sub-routine

Algorithm 3 gives a pseudo-code description of the PrivateMLE sub-routine. The algorithm is based on the "objective perturbation" framework developed in (Chaudhuri et al. 2011, Kifer et al. 2012). More specifically, Algorithm 3 calibrates a noisy term $(w^\top \theta)$ into the constrained maximum likelihood estimation formulation in order to achieve differential privacy of the output optimal solutions $\widehat{\theta}_n^p$.

The following proposition establishes the claim that Algorithm 3 is (ε, δ)-differentially private.

Proposition 4. *The output of Algorithm 3, $\widehat{\theta}_n^p$, satisfies (ε, δ)-differential privacy.*

The next corollary, which establishes the privacy guarantee of the PrivateMLE, immediately follows Proposition 4 and Corollary 1. It shows how to set the algorithmic parameters in Algorithm 3 to ensure that the resulting price decisions are differentially private at the designated levels ε and δ.

Corollary 3. *Suppose* PrivateMLE *is invoked for at most D_∞ times in Algorithm 1. Then the composite sequence of D_∞ outputs of* PrivateMLE *satisfies (ε, δ)-differential privacy if each call of* PrivateMLE *is supplied with privacy parameters $\delta' = \frac{\delta}{2D_\infty}$ and $\varepsilon' = \frac{\varepsilon}{\sqrt{2D_\infty \ln(1/\delta')}}$.*

Now, we are ready to provide the privacy guarantee of the entire policy in Algorithm 1.

Corollary 4. *The price decisions $\{p_1, \ldots, p_T\}$ of Algorithm 1 satisfy $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differential privacy.*

Corollary 4 immediately follows Proposition 3, Corollary 3, Proposition 2, and Fact 1. More specifically, in Algorithm 1, PrivateCov is invoked with parameters $(\varepsilon_1, \delta_1)$, which is $(\varepsilon_1, \delta_1)$-differential privacy. Moreover, PrivateMLE is invoked with parameters $(\varepsilon_2', \delta_2')$ for at most D_∞ times, whose outputs are $(\varepsilon_2, \delta_2)$-differential privacy. Therefore, the entire policy satisfies $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differential privacy, thanks to Proposition 2 and the basic composition rule in Fact 1.

In the rest of this section we establish the prediction error guarantee (a.k.a., the utility guarantee in differential privacy literature) of the estimator $\widehat{\theta}_n^p$ from the PrivateMLE sub-routine. More

Algorithm 3 The PrivateMLE sub-routine

```
function PrivateMLE($(n, \rho, \varepsilon, \delta)$)                                    ▷ returns $\widehat{\theta}_n^p$
    $B_1 \leftarrow (B_Y+1)G,\ B_2 \leftarrow KG,\ \rho \leftarrow \max\{\rho, 2B_2/\varepsilon\},\ \nu_{\varepsilon,\delta}^2 \leftarrow B_1^2(8\ln(2/\delta)+4\varepsilon)/\varepsilon^2$;
    Sample $w \sim \mathcal{N}(0, \nu_{\varepsilon,\delta}^2 I_d)$;
    Return $\widehat{\theta}_n^p = \arg\min_{\|\theta\|_2 \le 2}\{(\sum_{t<n} -\ln p(y_t|\phi_t,\theta)) + \frac{\rho}{2}\|\theta\|_2^2 + w^\top\theta\}$;
end function
```

precisely, Lemma 2 below upper bounds the prediction errors of the sequence of obtained model estimate $\widehat{\theta}_n^p$ with the presence of artificially calibrated noises in the PrivateMLE sub-routine. With smaller values of ε, δ indicating stronger privacy protection, the parameter $\nu_{\varepsilon,\delta}$ becomes larger, which leads to a larger variance of the Gaussian noise w. Thus, the PrivateMLE sub-routine needs to calibrate higher magnitudes of noise into the objective function, leading to either larger prediction errors (see (4)) or lengthened forced exploration phase (see the condition of (5)). Note that the lower bound on $\lambda_{\min}(\Sigma_n)$ for (5) is achieved by the exploration phase in Algorithm 1. This key result quantifies the tradeoff between the strength of the privacy protection and the prediction errors of the model parameter estimates. Another practical guidance from Lemma 2 is that the regularization amount ρ also needs to grow as ε, δ becomes smaller.

We emphasize that this key utility guarantee in Lemma 2 is *not* directly covered by existing utility analysis in Chaudhuri et al. (2011), Kifer et al. (2012) for two reasons. First, in Chaudhuri et al. (2011), Kifer et al. (2012), the utility is measured in terms of the difference between *objective values* before and after objective perturbation, which is *not* sufficient for the purpose of analyzing contextual bandit algorithms that require first-order KKT conditions. Additionally, in both Chaudhuri et al. (2011), Kifer et al. (2012), the data (ϕ_t, y_t) are assumed to be sampled *independently and identically* from an underlying distribution, while in our problem the data clearly are neither independent nor identically distributed.

We also remark that our utility analysis of the (differentially private) constrained maximum likelihood estimation (see the proof of Lemma 2) differs significantly from existing analysis of generalized linear contextual bandit problems as well (Filippi et al. 2010, Li et al. 2017, Wang et al. 2019). In Li et al. (2017), it is assumed that ϕ_t are *i.i.d.* and their distributions satisfy a certain non-degenerate assumption, which we do not necessarily impose in this paper. In both Filippi et al. (2010) and Wang et al. (2019), the formulations of the optimization problems are non-convex in θ, which facilitates the analysis of the properties of the optimal solution. However, the non-convex formulation poses significant challenges for privacy-aware algorithms since differentially private methods for non-convex optimization are scarce. It is therefore a highly non-trivial task to analyze a fully convex optimization formulation without stochasticity assumptions on ϕ_t.

LEMMA 2. *Fix* $n \in \{1, 2, \cdots, T-1\}$ *and let* $\Lambda_n = \Sigma_n + \rho I = \sum_{t<n} \phi_t \phi_t^\top + \rho I$*. Suppose* $\rho \geq \max\{5\nu_{\varepsilon,\delta}\sqrt{5d\ln T}, 2 + 48s^2G^2Kd\ln T\}$*. Then with probability* $1 - O(T^{-2})$ *the following hold:* $\|\widehat{\theta}_n^p\|_2 < 2$*, and*

$$(\widehat{\theta}_n^p - \theta^*)^T \Lambda_n (\widehat{\theta}_n^p - \theta^*) \leq \left(sK\sqrt{3d\ln T} + (2G+3)\sqrt{\rho} + G\nu_{\varepsilon,\delta}\sqrt{5d\ln T}\right)^2. \tag{4}$$

Furthermore, if $\lambda_{\min}(\Sigma_n) \geq \lambda_0 = [\frac{(2G+3)\rho}{\sqrt{5d\ln T}} + \nu_{\varepsilon,\delta}G]^2$*, then the above inequality can be strengthened to*

$$(\widehat{\theta}_n^p - \theta^*)^T \Lambda_n (\widehat{\theta}_n^p - \theta^*) \leq [4sK\sqrt{d\ln T}]^2. \tag{5}$$

Lemma 2 is proved by analyzing the first-order KKT condition at $\widehat{\theta}_n^p$, and is deferred to the supplementary material. Lemma 2 upper bounds the transformed estimation error of the differentially private MLE $\widehat{\theta}_n^p$ in two upper bounds. The first upper bound in (4) applies to the general setting and has a $G\nu_{\varepsilon,\delta}\sqrt{5d\ln T}$ additive term involving the differential privacy parameters ε, δ. in the upper bound. The second upper bound in (5), on the other hand, shows that if the sample covariance matrix Σ_n is spectrally lower bounded, then the upper bound on $\|\widehat{\theta}_n^p - \theta^*\|_{\Lambda_n}^2$ can be much improved with only the standard $O(\sqrt{d\ln T})$ term.

7. Regret analysis

Section 6 has established the privacy guarantees of our dynamic personalized pricing policy (see Corollary 4). In this section, we will further analyze the *performance/utility* of our proposed policy by proving upper bounds on its expected cumulative *regret*.

Recall that in the dynamic personalized pricing problem, there are t time periods and at each time period a customer arrives with personal information x_t. When offered price p_t, the expected demand is modeled by the generalized linear model $p(y_t|p_t, x_t, \theta^*) = \exp\{\zeta(y\phi(p_t, x_t)^\top\theta^* - m(\phi(p_t, x_t)^\top\theta^*) + h(y_t, \zeta)\}$ with expectation $\mathbb{E}[y_t|p_t, x_t, \theta^*] = f(\phi(p_t, x_t)^\top\theta^*)$. With θ^* known in hindsight, the optimal price p_t^* at time t is the one maximizing the retailer's expected revenue, or more specifically

$$p_t^* := \arg\max_{p\in[0,1]} pf(\phi(p, x_t)^\top\theta^*).$$

The *regret* of a dynamic pricing policy π is then defined as the cumulative difference between the expected revenue of the policy's offered prices and that of a clairvoyant, or more specifically

$$\text{Regret}(\pi; T) := \sum_{t=1}^{T} p_t^* f(\phi(p_t^*, x_t)^\top\theta^*) - p_t f(\phi(p_t, x_t)^\top\theta^*).$$

Clearly, by definition, the regret of any admissible policy is always non-negative since no p_t has a higher expected revenue compared to p_t^*. The smaller the regret, the better the policy's performance. We are also primarily focused on the *asymptotic* growth of the regret as a function of the time horizon T, as well as several other important parameters, such as the feature dimension d and the privacy parameters $\varepsilon_0 := \varepsilon_1 + \varepsilon_2$, $\delta_0 := \delta_1 + \delta_2$.

7.1. The general case

We first analyze the regret of Algorithm 1 in the most general case, in which customers' personal information $\{x_t\}$ is obliviously (i.e., pre-fixed) but can be adversarially chosen without pre-assumed patterns. Our next theorem upper bounds the regret of Algorithm 1 with proper choices of the values of algorithmic parameters. Recall that $\varepsilon_0 := \varepsilon_1 + \varepsilon_2$, $\delta_0 := \delta_1 + \delta_2$. We also note that for the general case, the random exploration phase (Step 4 in Algorithm 1) will be unnecessary and thus we could set $T_0 = 0$.

THEOREM 1. *Suppose Algorithm 1 is run with parameters* $\varepsilon_1, \varepsilon_2 \geq 0.1\varepsilon_0$, $\delta_1, \delta_2 \geq 0.1\delta_0$, $T_0 = 0$, $D_\infty = \lceil d \log_{1.5} T \rceil$, $\rho = \max\{\varepsilon_1^{-1} d^{1.5} \ln^5 T, 5\nu_{\varepsilon_2', \delta_2'}\sqrt{5d\ln T}, 2 + 48s^2G^2Kd\ln T\}$, $\gamma = K[(\sqrt{3}sK + \sqrt{5}G\nu_{\varepsilon_2', \delta_2'})\sqrt{d \ln T} + (2G+3)\sqrt{\rho}]$, *where* $\varepsilon_2', \delta_2'$ *are defined in Step 3 of Algorithm 1 and* $\nu_{\varepsilon_2', \delta_2'}$ *is defined in Algorithm 3. Then it holds that*

$$\mathrm{Regret}(\pi; T) \leq 2\gamma\sqrt{4.6dT\ln T} \leq \widetilde{O}\left(\varepsilon_0^{-1}\sqrt{d^3 T \ln^5(1/\delta_0)}\right),$$

where in the $\widetilde{O}(\cdot)$ *notation we omit logarithmic terms in* T *and polynomial dependency on other model parameters* s, K, G *and* B_Y*.*

Theorem 1 is proved in the supplementary material. We note that when T is large, our regret bound matches the classical optimal regret bound of $O(\sqrt{T})$. The dependency on the dimensionality of personal information d (i.e., $\sqrt{d^3}$) can be further improved by assuming a stronger assumption on the stochasticity of personal information x_t (see Section 7.2). Stochastic personal information or demand covariate has been a common assumption in the pricing literature (see e.g., Qiang & Bayati 2016, Ban & Keskin 2021, Javanmard & Nazerzadeh 2019, Chen et al. 2021).

7.2. Improved regret with stochastic contexts

In this section, we show that for a large class of problems in which the customers' personal information is *stochastically distributed*, the regret upper bound in Theorem 1 could be significantly sharpened.

The following assumption mathematically characterizes the stochasticity condition of customers' personal information used in this section:

ASSUMPTION 1. *Let $U[0,1]$ be the uniform distribution on $[0,1]$. There exists an underlying distribution μ_x and a constant $\kappa_x > 0$ such that, $x_1, \cdots, x_T \overset{i.i.d.}{\sim} \mu_x$, and furthermore*

$$\|\phi(x,p)\|_2 \leq 1 \quad a.s. \;\sim \mu_x \times U[0,1]; \qquad \mathbb{E}_{(x,p)\sim\mu_x\times U[0,1]}\left[\phi(p,x)\phi(p,x)^\top\right] \succeq \kappa_x I_d.$$

Assumption 1 assumes that consumers' personal feature vectors are relatively widely spread, so that they are not concentrated in a narrow region or direction. Such an assumption helps improve the regret analysis because the algorithm can expect to see feature vectors along with any directions with reasonable chances, and therefore the overall estimates of the unknown regression model can be more accurate.

With Assumption 1, the following theorem shows that when algorithmic parameters are properly chosen in Algorithm 1, the regret upper bound can be improved compared to Theorem 1 for the general setting.

THEOREM 2. *Under Assumption 1, suppose Algorithm 1 is run with parameters $\varepsilon_1, \varepsilon_2 \geq 0.1\varepsilon_0$, $\delta_1, \delta_2 \geq 0.1\delta_0$, $D_\infty = \lceil d \log_{1.5} T \rceil$, $\rho = \max\{\varepsilon_1^{-1} d^{1.5} \ln^5 T, 5\nu_{\varepsilon_2', \delta_2'} \sqrt{5d \ln T}, 2 + 48s^2 GKd \ln T\}$, $T_0 = 32[\frac{(2G+3)\rho}{\sqrt{5d\ln T}} + \nu_{\varepsilon,\delta} G]^2 \ln^2(dT)$, $\gamma = 4sK^2\sqrt{d \ln T}$, where $\varepsilon_2', \delta_2'$ are defined in Step 3 of Algorithm 1 and $\nu_{\varepsilon_2', \delta_2'}$ is defined in Algorithm 3. Then it holds for sufficiently large $T \geq e^{\kappa_x^{-2}}$ that*

$$\mathrm{Regret}(\pi, T) \leq T_0 + 2\gamma\sqrt{4.6dT\ln T} \leq \widetilde{O}\left(d\sqrt{T} + \varepsilon_0^{-2} d^2 \ln^{10}(1/\delta_0)\right),$$

where in the $\widetilde{O}(\cdot)$ notation we omit logarithmic terms in T and polynomial dependency on other model parameters s, K, G and B_Y.

The proof of Theorem 2 is largely the same as the proof of Theorem 1, except for the application of the second upper bound in Lemma 2. We relegate the complete proof of Theorem 2 to the supplementary material. Comparing Theorem 2 with Theorem 1, we note that the significant improvement lies in the additive nature between ε_0, δ_0 and d, T terms in Theorem 2. More specifically, because the privacy-incurred terms are now additive and do not scale polynomially with T, in most practical scenarios when the time horizon T is very large, the dominating term of Theorem 2 becomes only $\widetilde{O}(d\sqrt{T})$, which is optimal (up to logarithmic factors) in *both* the time horizon T and the feature dimension d (see, for example, the $\Omega(d\sqrt{T})$ lower bound in Dani et al. (2008)).

7.3. Impact of privacy constraints on seller surplus

In our theoretical framework, the seller surplus is measured and reflected by the notion of *regret*, which measures how much revenue/profits are lost by the seller's pricing decisions compared to the optimal personalized prices in hindsight. The smaller the regret, the larger the seller surplus.

Our main results in Theorems 1 and 2 give quantitative upper bounds on the regret of our proposed algorithm. More specifically, the regret of our algorithm is (omitting logarithmic factors and secondary model parameters) $\widetilde{O}(\varepsilon^{-1}\sqrt{d^3T})$ in the general setting, and $\widetilde{O}(\sqrt{d^2T}+\varepsilon^{-2}d^2)$ with additional assumptions on the distribution of consumers' context vectors. Here T is the time horizon (i.e., the number of customers handled), d is the number of covariates in consumers' personal data, and $\varepsilon > 0$ dictates the level of privacy leakage, with smaller ε indicating stronger/stricter protection of users' privacy. Based on these results, we make the following observations:

Tradeoffs between seller profits and privacy protection. With stronger privacy protection (i.e., $\varepsilon \to 0^+$), it is clear that the regret of our proposed algorithm increases, indicating that the seller profits are going to suffer with additional privacy constraints. The decrease of seller surplus is, however, alleviated when the consumers' context vectors are relatively well distributed, as the $\varepsilon^{-2}d^2$ term is *not* the dominating term in the regret bound when there are sufficient number of customers/users. Such decrease of seller profits is intuitive and expected, because additional privacy constraints limit sellers' ability to offer very personally tailored prices to boost their revenues.

Value and privacy costs of information. The d parameter in the regret bound characterizes how many covariates or factors the pricing algorithm exploits in customers' personalized data and shows the value and privacy costs of information: with more factors/covariates (i.e., larger values of d), the retailer is able to consider more refined details and information of each incoming customer but such information adds to the burden of privacy protection, leading to increased regret. To see this more clearly, with some stochasticity assumption of covariates, the regret bound $\widetilde{O}\left(d\sqrt{T}+\varepsilon^{-2}d^2\right)$ in Theorem 2 shows the following fact. For the regret term $\varepsilon^{-2}d^2$ related to the privacy to be a constant, a larger dimension d (i.e., more customer information) implies that $\varepsilon = C_0 d$ also grows proportionally, which leads to a weaker privacy protection. Additionally, for the first term $\widetilde{O}(d\sqrt{T})$, there is also a known lower bound showing that any policy must suffer a regret of $\Omega(d\sqrt{T})$ in the worst case (Dani et al. 2008). Therefore, there is indeed a cost of information for the purpose of privacy protection. Our regret upper bounds therefore provide in principle a bottom line for practitioners to gauge the costs of incorporating more factors of user information into dynamic personalized price decisions.

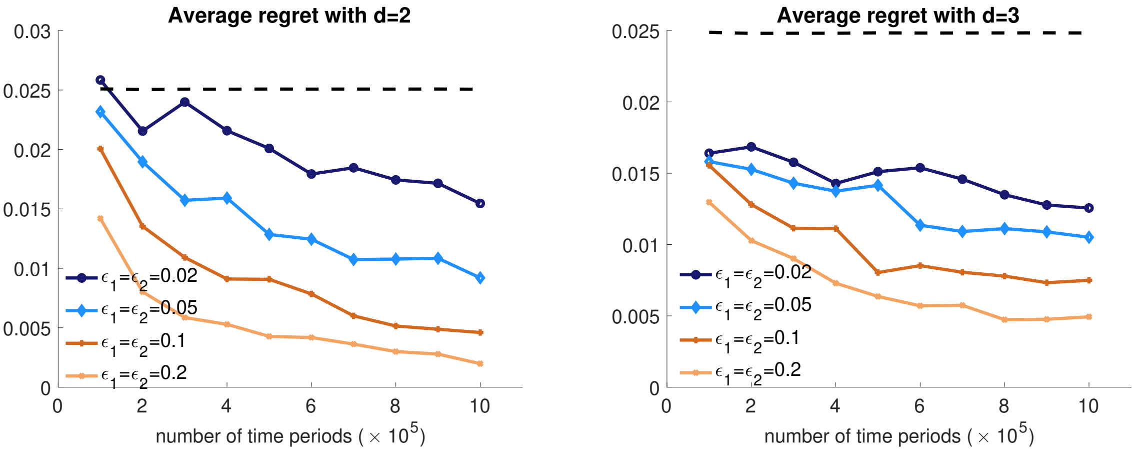

Figure 6 Average regret of our proposed algorithm under different time horizons T. The black dashed line indicates the average regret of a policy offering completely at random prices. Both δ_1, δ_2 parameters are set at $\delta_1 = \delta_2 = 1/T^2$.

8. Numerical results

In this section we corroborate the theoretical guarantees established in this paper for our proposed differentially private personalized pricing method with simulation results on a synthetic dataset. We adopt the logistic regression model $\Pr[y_t = 1|\phi_t, \theta^*] = \frac{e^{\zeta\phi_t^\top\theta^*}}{1+e^{\zeta\phi_t^\top\theta^*}}$, with $\zeta = 4$, $\phi_t(x_t, p_t) = \frac{1}{\sqrt{d}}[x_t; -p_t] \in \mathbb{R}^d$ and $\theta^* = [-\sqrt{0.1}; -\sqrt{0.1}; \cdots; -\sqrt{0.1}; \sqrt{1-0.1(d-1)}] \in \mathbb{R}^d$. The personal feature vectors $\{x_t\}$ are synthesized uniformly at random from the unit cube $[-1,1]^{d-1}$. It is easy to verify that $\|\phi_t\|_2 \le 1$ and $\|\theta^*\|_2 \le 1$ always hold for all d. Algorithm parameters (as inputs in Algorithm 1) are chosen as $T_0 = 10$, $\rho = 10$, $D_\infty = \lceil d \log_2 T \rceil$, and $\gamma = 1$. Other privacy-related parameters will be varied to demonstrate a spectrum of our proposed algorithm on a continuous landscape of differential privacy guarantees. Note that this experiment's main purpose is to investigate the impact of privacy-related parameters (i.e., ε and δ) rather than compete with state-of-the-art non-private pricing algorithms.

In Figure 6 we plot the average regret of our proposed algorithm under various $\varepsilon_1, \varepsilon_2$ privacy settings and time horizons T ranging from 10^5 to 10^6. All settings are run for 20 independent trials and the average regret is reported. For reference purposes, we also indicate in both plots of Figure 6 (see the flat dashed line) the average regret of a policy that simply produces uniformly at random prices p_t at each t, completely ignoring the personalized features/factors of each incoming customer. As we can see, under most privacy settings including highly secured settings with small ε (e.g., $\varepsilon_1 = \varepsilon_2 = 0.02$), the average regret of our proposed algorithm is much smaller compared to completely random prices, demonstrating its utility under privacy constraints. Furthermore, with relaxed privacy requirements (i.e., larger values of $\varepsilon_1, \varepsilon_2$) and/or longer pricing horizons T, the average regret of our algorithm significantly decreases, which verifies the theoretical regret upper bounds we established in Theorems 1 and 2.

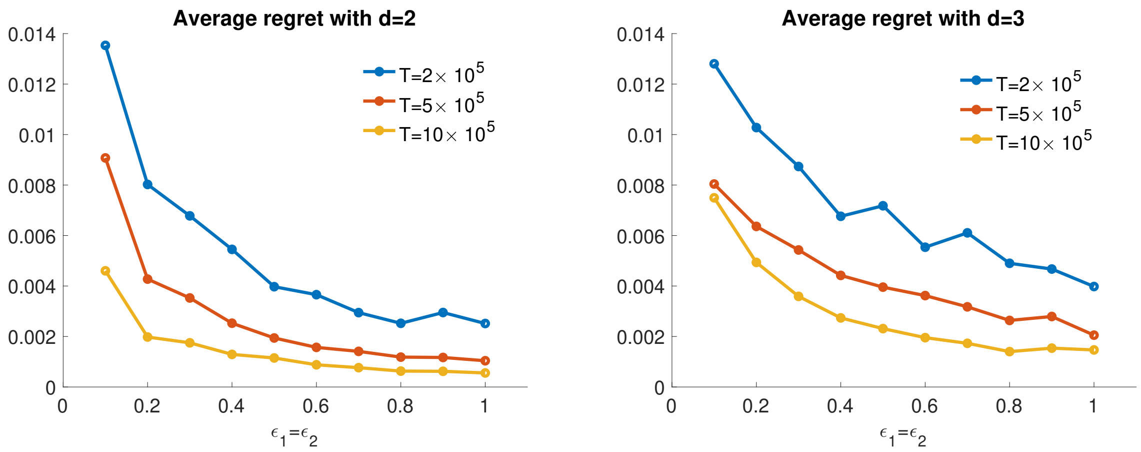

Figure 7 **Average regret of our proposed algorithm under different privacy parameters $\varepsilon = \varepsilon_1 = \varepsilon_2$. Both δ_1, δ_2 parameters are set at $\delta_1 = \delta_2 = 1/T^2$.**

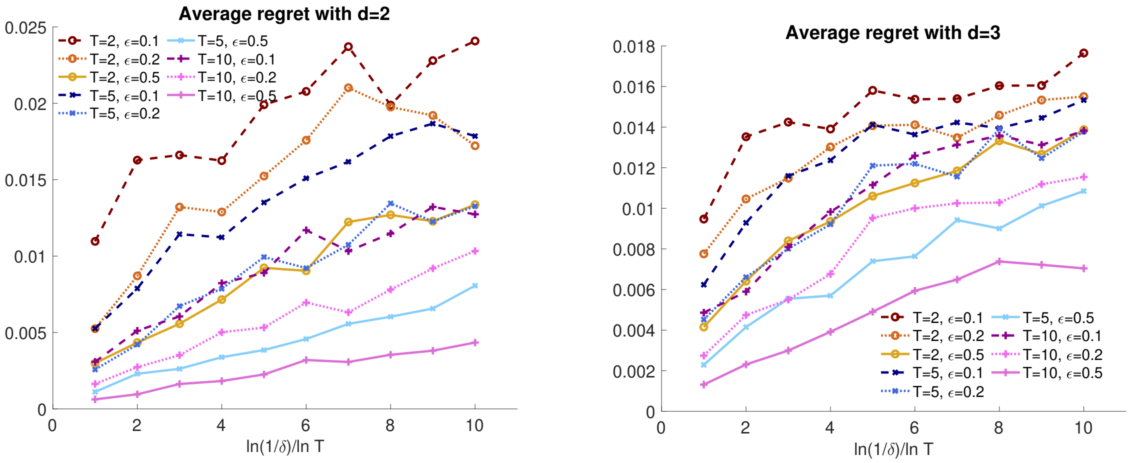

Figure 8 **Average regret of our proposed algorithm under different δ parameter values. From left to right the δ values are $1/T, 1/T^2, \cdots, 1/T^{10}$. The time horizon is measured in terms of 10^5 periods (i.e., $T = 2$ means 2×10^5 total time periods).**

In Figures 7 and 8, we provide some additional auxiliary simulation results. Figure 7 gives a direct landscape of the average regret of our algorithm under ε values ranging from 0.1 to 1. Figure 8 further explores the robustness of our algorithm under several very small δ values (as small as $\delta = 1/T^{10}$). Note that in Figure 8 there are multiple trend lines corresponding to the performances of the proposed algorithm under different settings of T, ε and δ values. Apart from the dependency on $\ln(1/\delta)$, Figure 8 also shows that the average regret of our algorithm decreases with increasing time horizon T and relaxed privacy guarantees (i.e., larger values of ε), both of which are consistent with the findings in Figures 6 and 7. The results in both figures are as expected (significant decreases in average regret with large ε values and moderate increases in average regret with geometrically decreasing δ values) from our theoretical results.

Table 1 Average regret comparison with non-private pricing algorithms.

	$\varepsilon = 0.1$	$\varepsilon = 0.2$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 5.0$	non-private
$T = 10^5, d = 2$	201×10^{-4}	142×10^{-4}	74.6×10^{-4}	41.9×10^{-4}	44.7×10^{-4}	3.1×10^{-4}
$T = 10^6, d = 2$	46.0×10^{-4}	19.8×10^{-4}	11.5×10^{-4}	5.6×10^{-4}	5.5×10^{-4}	0.6×10^{-4}
$T = 10^5, d = 3$	156×10^{-4}	130×10^{-4}	92.6×10^{-4}	62.9×10^{-4}	43.4×10^{-4}	3.1×10^{-4}
$T = 10^6, d = 3$	74.9×10^{-4}	49.4×10^{-4}	23.1×10^{-4}	14.7×10^{-4}	5.7×10^{-4}	1.6×10^{-4}

Table 2 Average regret with $T = 10^5$ of our algorithm under different $\varepsilon_1, \varepsilon_2$ settings. When the row indicates "fix $\varepsilon_1 \equiv 0.1$" (or "fix $\varepsilon_2 \equiv 0.1$"), then the ε in the column represents the value of ε_2 (or accordingly ε_1).

	$\varepsilon = 0.02$	$\varepsilon = 0.05$	$\varepsilon = 0.1$	$\varepsilon = 0.2$	$\varepsilon = 0.5$
$T = 10^5, d = 2$, fix $\varepsilon_1 \equiv 0.1$	0.0247	0.0232	0.0195	0.0145	0.0073
$T = 10^5, d = 2$, fix $\varepsilon_2 \equiv 0.1$	0.0192	0.0178	0.0196	0.0192	0.0191
$T = 10^5, d = 3$, fix $\varepsilon_1 \equiv 0.1$	0.0164	0.0160	0.0147	0.0128	0.0092
$T = 10^5, d = 3$, fix $\varepsilon_2 \equiv 0.1$	0.0145	0.0149	0.0144	0.0149	0.0154

To better illustrate our algorithm, we further report two additional sets of simulation results. In Table 1 we report the average regret of our proposed algorithm together with an algorithm that is not subject to any kind of privacy constraints, which is implemented by removing all noise calibration steps in the two private releasers PrivateCov and PrivateMLE. We remark that even larger ε values (e.g., $\varepsilon = 1.0$ or $\varepsilon = 5.0$) indicate quite non-trivial universal privacy protection of consumers' sensitive data, which explains the relatively larger regret incurred by differentially private pricing algorithms compared with their non-private counterparts. In Table 2 we report the average regret of our proposed algorithm when the values of ε_1 and ε_2 are very different to see which privacy parameter has a bigger impact on the performance of the designed algorithm. Table 2 shows that ε_2 clearly has a much larger effect on the regret performance of our algorithm, with the average regret significantly decreasing with larger ε_2 values. On the other hand, the impact of ε_1 is not significant or clear. This is expected from the structure of the algorithm, because ε_2 is used in the PrivateMLE sub-routine, which directly affects the model estimates used in subsequent price offerings.

9. Discussion and insights

9.1. Insufficiency of input perturbation

Input perturbation is a straightforward method for designing differentially private algorithms and is actually an effective method in some application scenarios. The high-level idea of input perturbation is to artificially calibrate noise directly to the *inputs* of the algorithm in order to protect private information. With noisy inputs, the privacy of the entire algorithm trivially follows from the closeness-to-post-processing property of differential privacy (Fact 2).

Table 3 **Average regret of our proposed algorithm and the input perturbation method.**

	our algorithm			input perturbation		
	$\varepsilon=0.2$	$\varepsilon=0.5$	$\varepsilon=1.0$	$\varepsilon=0.2$	$\varepsilon=0.5$	$\varepsilon=1.0$
$T=10^5, d=2\ (\times 10^{-4})$	142	74.6	41.9	393	393	98.2
$T=5\times 10^5, d=2\ (\times 10^{-4})$	42.7	19.4	10.4	393	393	95.8
$T=10^6, d=2\ (\times 10^{-4})$	19.8	11.5	5.6	393	393	95.3

In the context of personalized dynamic pricing, application of the input perturbation method amounts to calibrating noise directly to the personal features x_t of each incoming customer: $\widetilde{x}_t = x_t + \omega_t$, for some centered noise vectors $\{\omega_t\}_{t=1}^T$. Such an approach, however, is likely to fail because the features of each individual customer are relatively independent from each other. Therefore, a very large magnitude of noises $\{\omega_t\}$ need to be injected, which renders the subsequent pricing algorithm impractical. More detailed discussion follows:

1. Suppose $\widetilde{x}_t = x_t + w_t$ is the anonymized version of a customer's feature vector x_t at time t. Because $\widetilde{x}_t$ is released and used in the subsequent process of the pricing algorithm, one must ensure that $\widetilde{x}_t$ is differentially private. This means that the magnitude of w_t must be sufficiently large (on the order of $\Omega(1/\varepsilon)$) to protect the sensitive information of x_t.
2. Usually, input perturbation results in a much worse performance of the differentially private algorithms compared to output perturbation. Consider the very simple example of having sensitive data $x_1, \cdots, x_n$ and one wants to release $\overline{x} = \frac{1}{n}\sum_{i=1}^n x_i$ with ε-differential privacy. If we use input perturbation with $\widetilde{x}_i = x_i + w_i$ and the Laplace mechanism, we have $w_i \sim \mathrm{Lap}(0, 1/\varepsilon)$ and therefore $\widetilde{x}^1 := \frac{1}{n}\sum_{i=1}^n \widetilde{x}_i$ satisfies $\mathbb{E}[|\widetilde{x}^1 - \overline{x}|] \asymp O(1/\varepsilon\sqrt{n})$. On the other hand, if one uses output perturbation by releasing $\widetilde{x}^2 := \overline{x} + \frac{1}{n}\mathrm{Lap}(0, 1/\varepsilon)$, then one has $\mathbb{E}[|\widetilde{x}^2 - \overline{x}|] \asymp O(1/\varepsilon n)$. It is easy to verify that both $\widetilde{x}^1, \widetilde{x}^2$ are differentially private, but $\widetilde{x}^2$ clearly is much closer to $\overline{x}$ compared to $\widetilde{x}^1$. This very simple example shows that, in general, input perturbation (directly adding noises to sensitive data) is usually less efficient and should be avoided if there are better approaches.
3. In the particular model studied in this paper, the use of a generalized linear model further complicates the input perturbation-based methods. For many generalized linear models, such as the logistic regression model, the efficiency of statistical estimates (e.g., the maximum likelihood estimation) decays *exponentially* fast with respect to the vector norm of the feature vector x. Hence, if we use $\widetilde{x}_t = x_t + w_t$ to replace x_t directly in the logistic regression model, the norm of $\widetilde{x}_t$ is on the order of $\Omega(1/\varepsilon)$ and therefore the resulting method is going to incur an $O(\exp\{1/\varepsilon\})$ term in regret, which makes the regret excessively large.

In Table 3 we compare the average regret of our proposed algorithm with the input perturbation method using numerical simulations. Table 3 shows that the regret of our designed algorithm is

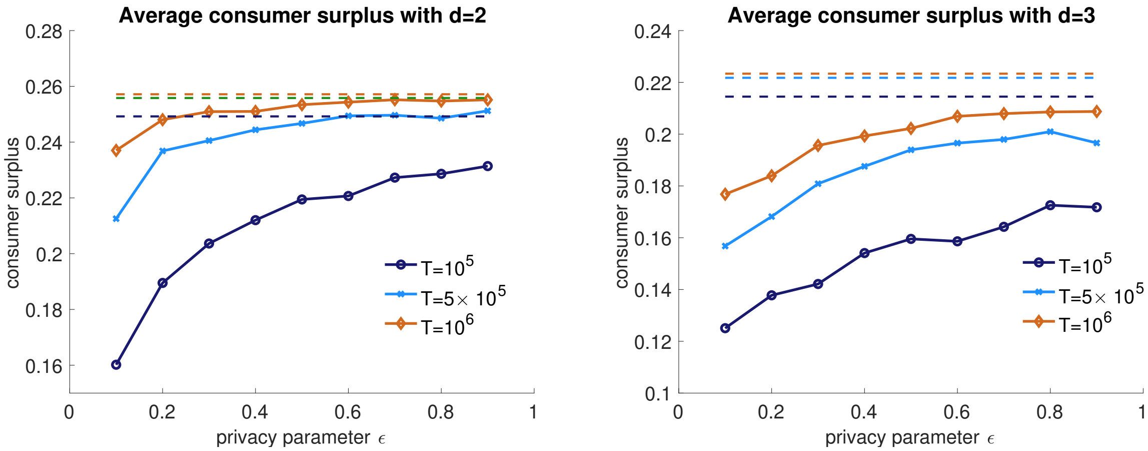

Figure 9 Average consumer surplus under different levels of privacy constraints and time horizons. The dashed lines represent average consumer surplus for a personalized pricing algorithm not subject to any data privacy constraints. Both $\varepsilon_1, \varepsilon_2$ parameters are equal to ϵ in the figures, and both δ_1, δ_2 parameters are set as $1/T^2$, where T is the time horizon.

significantly smaller than that of the input perturbation. Furthermore, the average regret of input perturbation is very large unless the ε parameter is at least one and does not necessarily decrease with increasing number of time periods T.

9.2. Impact of privacy constraints on consumer surplus

In this section we study the impact of privacy constraints of the seller's personalized pricing algorithm on the average consumer's surplus, under different levels of privacy constraints. We model the utility u_t for each incoming customer at time t with feature vector x_t and offered price p_t as $u_t = \zeta\langle\phi(x_t,p_t),\theta^*\rangle + \zeta_t$, where $\zeta = 4$, $\phi(x_t,p_t) = \frac{1}{\sqrt{d}}[x_t;p_t]$ and ζ_t are i.i.d. random variables following the standard centered Logistic distribution. The customer will make one unit of purchase if $u_t > 0$, resulting in a surplus of u_t, and leave without making any purchases if $u_t < 0$, resulting in zero surplus at that period. It is easy to verify that this utility model leads to the logistic regression model we used in the numerical experiments, or more specifically, $\Pr[y_t = 1|x_t,p_t] = \Pr[u_t > 0|x_t,p_t] = \frac{e^{\zeta\phi_t^\top\theta^*}}{1+e^{\zeta\phi_t^\top\theta^*}}$, where $\phi_t = \phi(x_t,p_t)$.

Figure 9 reports the average consumer surplus under our proposed privacy-aware personalized pricing algorithm, for both $d = 2$ and 3 with consumers' contextual vectors $\{x_t\}_{t=1}^T$ and the unknown regression model synthesized in the same way as in Section 8. We also plot the consumer surplus for a hypothetical pricing algorithm that is *not* subject to any privacy constraints as dashed lines in Figure 9. Note that we did not incorporate consumers' surplus from the protection of their private data, which is difficult to measure and compare against the surplus from their purchasing decisions. As we can see from Figure 9, as ε increases from 0 to 1, the implied privacy protection becomes *weaker* as the adversary has a stronger ability to distinguish between neighboring databases. This means that as ε increases, the seller has less ability to discriminate against customers based on their personal data and features, resembling a transition from first-degree to third-degree price discrimination. As a result, the consumer surplus increases as ε increases and the seller extracts less of the consumer surplus from his/her limited ability to carry out price discrimination.

10. Conclusions and future directions

In this paper, we investigate how to protect the privacy of a customer's personal information and purchasing decisions in personalized dynamic pricing with demand learning. Under the generalized linear model of the demand function, we propose a privacy-preserving constrained MLE policy. We establish both the privacy guarantee under the notion of anticipating differential privacy (DP) and the regret bounds for oblivious adversarial and stochastic settings.

There are several future directions. First, we could extend the current privacy setting to the local DP (Evfimievski et al. 2003, Kasiviswanathan et al. 2011), which is a stronger notion of DP. The local DP is suitable for distributed environments, as user terminals need to randomize data before sending it to the center. A very recent paper by Ren et al. (2020) investigates the UCB algorithm under the local DP. It would be interesting to study the personalized dynamic pricing under this stronger notion of DP. More importantly, as privacy has become a significant concern for the public, especially in the e-commerce domain, we believe that systematic research on privacy-preserving revenue management will become increasingly important in both academia and industry. While there is relatively less research in this area, we hope our work inspires future studies on privacy-aware operations management (e.g., inventory control or assortment optimization) based on the DP framework.

Acknowledgment

The authors thank the department editor, the associated editor, and the anonymous referees for many useful suggestions and feedback, which greatly improves the paper. Xi Chen is supported by National Science Foundation via the Grant IIS-1845444 and Facebook Faculty Research Award. David Simchi-Levi is supported by MIT-Accenture Alliance for Business Analytics and the MIT Data Science Lab.

References

Abowd, J. M. (2018). The u.s. census bureau adopts differential privacy. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD).*

Apple Differential Privacy Team (2017). Learning with privacy at scale. Tech. rep., Apple.

Araman, V. F., & Caldentey, R. (2009). Dynamic pricing for nonperishable products with demand learning. *Operations Research*, *57*(5), 1169–1188.

Ban, G.-Y., & Keskin, N. B. (2021). Personalized dynamic pricing with machine learning: High dimensional features and heterogeneous elasticity. *Management Science (forthcoming)*.

Barbaro, M., & Zeller, T. (2006). A face is exposed for aol searcher No. 4417749. The New York Times. https://www.nytimes.com/2006/08/09/technology/09aol.html.

Besbes, O., & Zeevi, A. (2009). Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research*, *57*(6), 1407–1420.

Besbes, O., & Zeevi, A. (2015). On the (Surprising) Sufficiency of Linear Models for Dynamic Pricing with Demand Learning. *Management Science*, *61*(4), 723–739.

Broder, J., & Rusmevichientong, P. (2012). Dynamic pricing under a general parametric choice model. *Operations Research*, *60*(4), 965–980.

Chan, T.-H. H., Shi, E., & Song, D. (2011). Private and continual release of statistics. *ACM Transactions on Information and System Security*, *14*(3), 1–24.

Chaudhuri, K., Monteleoni, C., & Sarwate, A. D. (2011). Differentially private empirical risk minimization. *Journal of Machine Learning Research*, *12*(3), 1069–1109.

Chen, L., Mislove, A., & Wilson, C. (2016). An empirical analysis of algorithmic pricingon amazon marketplace. In *Proceedings of the International Conference on World Wide Web*.

Chen, Q., Jasin, S., & Duenyas, I. (2015). Real-time dynamic pricing with minimal and flexible price adjustment. *Management Science*, *62*(8), 2437–2455.

Chen, X., Owen, Z., Pixton, C., & Simchi-Levi, D. (2021). A statistical learning approach to personalization in revenue management. *Management Science (forthcoming)*.

Cheung, W. C., & Simchi-Levi, D. (2017). Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. *Available at SSRN 3075658*.

Cheung, W. C., Simchi-Levi, D., & Wang, H. (2017). Dynamic pricing and demand learning with limited price experimentation. *Operations Research*, *65*(6), 1722–1731.

Cohen, M. C., Lobel, I., & Paes Leme, R. (2020). Feature-based dynamic pricing. *Management Science*, *66*(11), 4921–5484.

Dani, V., Hayes, T. P., & Kakade, S. M. (2008). Stochastic linear optimization under bandit feedback. In *Annual Conference on Learning Theory*.

den Boer, A. V., & Zwart, B. (2013). Simultaneously learning and optimizing using controlled variance pricing. *Management Science*, *60*(3), 770–783.

Ding, B., Kulkarni, J., & Yekhanin, S. (2017). Collecting telemetry data privately. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Dong, J., Roth, A., & Su, W. J. (2019). Gaussian differential privacy. *arXiv preprint arXiv:1905.02383v33*.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., & Naor, M. (2006a). Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*.

Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006b). Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference (TCC)*.

Dwork, C., Naor, M., Pitassi, T., & Rothblum, G. N. (2010). Differential privacy under continual observation. In *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*.

Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, *9*(3-4), 211–407.

Dwork, C., Talwar, K., Thakurta, A., & Zhang, L. (2014). Analyze Gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*.

Erlingsson, U., Pihur, V., & Korolova, A. (2014). Rappor: Randomized aggregatable privacy-preservingordinal response. In *Proceedings of the ACM SIGSAC Conference on Computer and Communication Security*.

Evfimievski, A., Gehrke, J., & Srikant, R. (2003). Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS)*.

Farias, V. F., & Van Roy, B. (2010). Dynamic pricing with a prior on market response. *Operations Research*, *58*(1), 16–29.

Ferreira, K. J., Simchi-Levi, D., & Wang, H. (2018). Online network revenue management using thompson sampling. *Operations Research*, *66*(6), 1586–1602.

Filippi, S., Cappe, O., Garivier, A., & Szepesvári, C. (2010). Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Hannak, A., Soeller, G., Lazer, D., Mislove, A., & Wilson, C. (2014). Measuring price discrimination and steering on e-commerce web sites. In *Proceedings of the The Internet Measurement Conference*.

Harrison, J. M., Keskin, N. B., & Zeevi, A. (2012). Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, *58*(3), 570–586.

Javanmard, A., & Nazerzadeh, H. (2019). Dynamic pricing in high-dimensions. *Journal of Machine Learning Research*, *20*(9), 1–49.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., & Smith, A. (2011). What can we learn privately? *SIAM Journal on Computing*, *40*(3), 793–826.

Kifer, D., Smith, A., & Thakurta, A. (2012). Private convex empirical risk minimization and high-dimensional regression. In *Annual Conference on Learning Theory*.

Lei, Y. M., Miao, S., & Momot, R. (2020). Privacy-preserving personalized revenue management. *Available at SSRN 3704446*.

Li, L., Lu, Y., & Zhou, D. (2017). Provably optimal algorithms for generalized linear contextual bandits. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, *7*(1), 76–80.

Miao, S., Chen, X., Chao, X., Liu, J., & Zhang, Y. (2019). Context-based dynamic pricing with online clustering. *arXiv preprint arXiv:1902.06199*.

Mishra, N., & Thakurta, A. (2015). (Nearly) optimal differentially private stochastic multi-arm bandits. In *In Proceedings of the Conference on Uncertainty in Artificial Intelligence*.

Mohammed, R. (2017). How retailers use personalized prices to test what you're willing to pay. *Harvard Business Review*.

Narayanan, A., & Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *Proceedings of the IEEE Symposium on Security and Privacy*.

Qiang, S., & Bayati, M. (2016). Dynamic pricing with demand covariates. *Available at SSRN 2765257*.

Ren, W., Zhou, X., Liu, J., & Shroff, N. B. (2020). Multi-armed bandits with local differential privacy. *arXiv preprint arXiv:2007.03121*.

Shariff, R., & Sheffet, O. (2018). Differentially private contextual linear bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Tang, W., Ho, C.-J., & Liu, Y. (2020). Differentially private contextual dynamic pricing. In *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems*.

Tringale, M. (2018). Dynamic pricing vs. personalized pricing, what's the difference? https://blog.wiser.com/dynamic-pricing-vs-personalized-pricing-whats-the-difference/.

Tsitsiklis, J., Xu, K., & Xu, Z. (2020). Private sequential learning. *Operations Research (to appear)*.

Wang, Y., Chen, X., Chang, X., & Ge, D. (2021). Uncertainty quantification for demand prediction in contextual dynamic pricing. *Production and Operations Management (forthcoming)*.

Wang, Y., Wang, R., Du, S. S., & Krishnamurthy, A. (2019). Optimism in reinforcement learning with generalized linear function approximation. *arXiv preprint arXiv:1912.04136*.

Wang, Y.-X. (2019). Per-instance differential privacy. *Journal of Privacy and Confidentiality*, *9*(1).

Wang, Y.-X., Lei, J., & Fienberg, S. E. (2016). On-average KL-privacy and its equivalence to generalization for max-entropy mechanisms. In *Proceedings of the International Conference on Privacy in Statistical Databases*.

Wang, Z., Deng, S., & Ye, Y. (2014). Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research*, *62*(2), 219–482.

Xu, J., Xu, K., & Yang, D. (2020). Optimal query complexity for private sequential learning against eavesdropping. *arXiv preprint arXiv:1909.09836*.

Xu, K. (2018). Query complexity of Bayesian private learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Zheng, K., Cai, T., Huang, W., Li, Z., & Wang, L. (2020). Locally differentially private (contextual) bandits learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.