Differentially Private Contextual Dynamic Pricing

Wei Tang * Chien-Ju Ho † Yang Liu ‡

Abstract

In this paper, we design differentially private algorithms for the contextual dynamic pricing problem. In contextual dynamic pricing, the seller sells heterogeneous products to buyers that arrive sequentially. At each time step, a buyer arrives with interests in purchasing a product. Each product is represented by a set of product features, i.e., the *context*, and the buyer's valuation for the product is a function of the product features and the buyer's private preferences. The goal of contextual dynamic pricing is to adjust the price over time to learn how to set the optimal price for the population from interacting with individual buyers. In the meantime, this learning process creates potential privacy concerns for individual buyers. A third-party agent might be able to infer the information of individual buyers from how the prices change after the participation of a particular buyer. In this work, using the notion of differential privacy as our privacy measure, we explore the design of differentially private dynamic pricing algorithms. The goal is to maximize the seller's payoff, or equivalently, minimize the *regret* with respect to the optimal policy when knowing the distribution of buyers' preferences while ensuring the amount of privacy leak of individual buyers' valuations is bounded. We present an algorithm that is ϵ-*differentially private* and achieves expected regret $\tilde{\mathcal{O}}\big(\frac{\sqrt{dT}}{\epsilon}\big)$, where d is the dimension of product features and T is the time horizon.

1 Introduction

Consider the pricing problem for an online retailer with products to sell. If the retailer/seller has full information about the demand curve, i.e., the distribution of buyers' valuations for her product, she[1] can calculate the optimal price that maximizes her own payoff (e.g., the price multiplied by the number of products sold). However, the information about the demand curve is often unknown a priori, and online retailers increasingly resort to adopt *dynamic pricing* strategies, which adaptively adjust prices to simultaneously gain information about the demand curve and maximize her own payoff. For example, Amazon.com has been known to dynamically adjust the prices of their products to maximize the revenue (Chen et al., 2016).

In the setup of (non-contextual) dynamic pricing, at each time step, the seller posts a price to an arriving buyer with unknown valuation. If the buyer's valuation is higher than the posted price, a sale occurs and the seller collects the payment; otherwise, no sale occurs. The goal of the seller is to design a pricing strategy, that takes into account of the interactions of past buyers, to optimize her own payoff. This dynamic pricing problem has been well studied in the literature (Kleinberg and Leighton, 2003; Balcan et al., 2008; Broder and Rusmevichientong, 2012). In general, it is possible to design a pricing strategy that converges to the optimal price as if the demand curve is known. This implies that, the seller can aggregate the information collected from individual buyers and learn the optimal price for the population.

On the other hand, since prices are updated based on the information of past buyers, it leads to potential privacy concern for individual buyers. A third-party agent, even without observing whether a sale occurs

*Washington University in St. Louis; w.tang@wustl.edu

†Washington University in St. Louis; chienju.ho@wustl.edu

‡University of California, Santa Cruz; yangliu@ucsc.edu

[1]In this paper, we use "she" to address the seller/retailer and "he" to address the buyer.

or not, might be able to infer the valuations of individual buyers (which represent buyers' private personal preferences or financial status) from how the prices change after the participation of individual buyers. To make things even worse, users are often interacting with many online retailers and participating in many online platforms, all this data about individuals could add up and lead to significant privacy leak. To address this issue, we adopt the notion of differential privacy (Dwork et al., 2006; Chan et al., 2011; Dwork et al., 2010), which has been the gold standard notion both in academia and in industry, to formally quantify the amount of privacy leak in the pricing strategy. Intuitively speaking, a common approach to improve privacy is to add *noise* in the data. However, this would simultaneously lead to the decrease of utility. The focus of differential privacy research have been to formalize the trade-off of privacy and utility.

In this work, we explore the above trade-off of privacy and utility in a *contextual* dynamic pricing problem. In contextual dynamic pricing, the seller has heterogeneous products to sell. Each product is represented by a set of public product features. The buyer's valuation for a product is a function of the public product feature and the buyer's private preferences. When each buyer arrives with interests in a certain product (the product features is the "context"), the seller posts a price for the product. The goal of the seller is to design a dynamic pricing policy, that takes into account interactions of all past buyers, such that the policy (1) maximizes her own payoff, or equivalently, minimizes the *regret* with respect to the optimal policy when knowing the distribution of buyers' preferences while (2) ensuring the amount of privacy leak of individual buyers' valuations is bounded.

Our main result is a differentially private contextual dynamic pricing algorithm which has two desired properties. First, it is ϵ-differentially private, i.e., an adversary cannot learn too much about any individual buyer's information from observing the output of the pricing algorithm. Second, the algorithm achieves a regret of $\tilde{\mathcal{O}}(\sqrt{dT}/\epsilon)$,[2] where d is the dimension of product features and T is the time horizon.

To summarize our approaches, we first approximate the contextual dynamic pricing problem as a full information online learning problem Qiang and Bayati (2016); Javanmard and Nazerzadeh (2019); Javanmard (2017); Amin et al. (2014). In order to achieve the requirement of differentially privacy, we leverage the techniques of differentially private online learning (Thakurta and Smith, 2013) and Tree-Based Aggregation Protocol (Chan et al., 2011; Dwork et al., 2010). However, directly applying their methods does not work in our setting, since our problem leads to non-convex loss functions due to the nature of binary feedback (sale/no sale). To get around this issue, we propose an alternative problem formulation and show how to connect the results of the alternative formulation to our focused contextual dynamic pricing problem. Lastly, we analyze the privacy and regret guarantees of our approaches. To the best of our knowledge, our work is the first to address the privacy issues in dynamic pricing problems.

2 Related Work

Dynamic Pricing. The dynamic-pricing literature studies pricing algorithms in settings when the demand function is unknown (Kleinberg and Leighton, 2003; Babaioff et al., 2015; Broder and Rusmevichientong, 2012; Besbes and Zeevi, 2009; Trovò et al., 2018). An early work in non-parametric setting is by Kleinberg and Leighton (2003), where they model dynamic pricing problems as multi-armed bandit settings where each arm corresponds to a (discretized) posted price. They propose an algorithm which achieves $\mathcal{O}(\sqrt{T})$ regret where T denotes the length of the learning horizon. Another natural approach is to model the uncertainty about buyers' valuations using a set of parameters, and then estimate those parameters using classical inference methods such as maximum likelihood or least square estimation (Broder and Rusmevichientong, 2012; den Boer and Zwart, 2013; Bastani and Bayati, 2015). Our work is similar to this line of work, in which we

[2]We use $\tilde{\mathcal{O}}(\cdot)$ to disregard the logarithmic factors.

assume a parametric model for buyers' valuations and apply inferences. More specifically, the setting we consider builds on models with features/covariates (Qiang and Bayati, 2016; Javanmard and Nazerzadeh, 2019; Javanmard, 2017; Amin et al., 2014), where the buyer's context-based valuation is a linear function of unknown parameter, public product features, and the buyer's individual parameter (i.e., preference shock). A regret bound of $\mathcal{O}(\sqrt{T})$ is achieved in several recent works (Broder and Rusmevichientong, 2012; Javanmard, 2017; Javanmard and Nazerzadeh, 2019). Other closely related work to ours is by Cohen et al. (2016) and Lobel et al. (2018). In their setting, the buyers are homogeneous and the valuation functions are deterministic functions, while we model heterogeneous buyers through the notion of preference shocks. Our work differs from the dynamic pricing literature through explicitly addressing the privacy concerns. To our knowledge, we are the first to incorporate the notion of differential privacy in dynamic pricing.

Differential Private Online Learning. Differential privacy (Dwork et al., 2006) is a rigorous notion requiring that changing the data of only a single individual, or alternatively, of only a single attribute of an individual, has a negligible effect on computations done using this data. In online settings, while it seems challenging to ensure privacy guarantees since a change in a single time step may affect the outputs at all future steps, is has has been shown to be achievable with elegant designs. This problem was first considered by Dwork et al. (2010) and Chan et al. (2011), where the authors introduced the tree-based aggregation protocol for releasing the cumulative sums of vectors in a differentially private manner, while ensuring that the total amount of noise added for each cumulative sum is only poly-logarithmically dependent on the number of vectors. Jain et al. (2012) considered a more general problem. The authors developed algorithms to preserve (ϵ, δ)-differential privacy and achieve regret bounds of the order $\tilde{\mathcal{O}}(\frac{1}{\epsilon}\sqrt{T}\log(\frac{1}{\delta}))$. Our technique is related to the one by Thakurta and Smith (2013), which provided a modified Follow-the-Approximate-Leader template for online convex optimization that achieves a regret rate $\tilde{\mathcal{O}}(\frac{\sqrt{dT}}{\epsilon})$. Moreover, Cardoso and Cummings (2019) study algorithms for online submodular minimization that preserve differential privacy under full information feedback and bandit feedback. Another close antecedent to our setting is a recent work by Shariff and Sheffet (2018), where the authors study how to achieve a different privacy notion, joint differential privacy in contextual bandits, they design a private linear-UCB algorithm via the tree-based technique to ensure privacy. Our work differs from this line of work in that our setting exhibit specific feedback structure (i.e., the binary feedback) and their algorithm cannot be applied.

3 Preliminaries and Framework

Notations. For any two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d$, we use $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ to denote their inner product. For a vector $\boldsymbol{v}$, $\|\boldsymbol{v}\|_p$ is the L_p-norm of $\boldsymbol{v}$, i.e., $\|\boldsymbol{v}\|_p = (\sum_i |v_i|^p)^{1/p}$. When the subscript p is omitted, it is the L_2-norm. Suppose that $f : \mathcal{X} \to \mathbb{R}$ is a real-valued function whose domain is an arbitrary set $\mathcal{X}$. Let $\mathsf{supp}(f)$ be the support of f, i.e., the set of points in $\mathcal{X}$ where f is non-zero.

3.1 Contextual dynamic pricing framework

Consider a pricing problem faced by a seller with heterogeneous products to sell. At each time period $t = 1, 2, ..., T$, a buyer arrives with interests in purchasing a product. We assume the buyer arrival is stochastic and randomly drawn from some unknown distribution. The context sequence could be adversarial. Each product is represented by a vector of features (i.e., context) denoted by $\boldsymbol{x}_t \in \mathbb{R}^d$, which is publicly observable. The buyer's valuation for the product is a function of the product features and his own private preference. To simplify the presentation, we start by assuming the buyer's value of a product is a linear function of the product features $\boldsymbol{x}$ and the buyer's preference $\boldsymbol{\theta}$. We discuss the generalization to other function forms in Section 7.

In particular, we follow the standard styled formulation in contextual dynamic pricing (Qiang and Bayati, 2016; Javanmard and Nazerzadeh, 2019; Amin et al., 2014; Cohen et al., 2016). We write the valuation function as follows:

$$v_t(\boldsymbol{x}_t) = \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle + z_t, \tag{1}$$

where θ is a population-wide parameter unknown to the seller, and z_t is a scalar random variable and is called *preference shocks* for individual buyer at time t.

Preference shocks are assumed to be i.i.d. drawn from a zero-mean distribution over $\mathbb{R}$.[3] We denote its cumulative distribution function by F, and the corresponding density by $f(z) = F'(z)$. When it is clear from the context, we omit the subscript and denote $v_t(\boldsymbol{x}_t)$ as $v(\boldsymbol{x}_t)$ or v_t. Feature vectors $\boldsymbol{x}_t$s are observable, while model parameter $\boldsymbol{\theta}$ is a-priori unknown to the seller. Therefore, the buyer's valuation $v(\boldsymbol{x}_t)$ is also unknown to the seller.

At each period t, the seller posts a price p_t. If $p_t \leq v(\boldsymbol{x}_t)$, a sale occurs, and the seller collects revenue p_t. If the price is set higher than the market value, $p_t \geq v(\boldsymbol{x}_t)$, no sale occurs and no revenue is generated. The goal of the seller is to design a pricing policy that maximizes the collected revenue. Note that at each step, the seller has access to the previous feedbacks (sale/no sale) from the buyer and can use this information to adaptively adjust the current price.

Technical assumptions. Without loss of generality, we normalize $\boldsymbol{x}_t$ and $\boldsymbol{\theta}$ such that $\|\boldsymbol{x}_t\| \leq 1$, and $\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\| \leq 1\}$ and $\|\boldsymbol{\theta}\| \leq W$ for a known constant W. We denote by Θ the set of feasible parameters, namely, $\Theta = \{\boldsymbol{\theta} \in \mathbb{R}^d : \|\boldsymbol{\theta}\| \leq W\}$. We now make the following assumptions on the distribution of F:

Assumption 1. The function $F(v)$ is strictly increasing. Further, $F(v)$ and $1 - F(v)$ are log-concave in v.

Log-concavity is a common modeling choice in the economics literature (Bagnoli and Bergstrom, 2005; Babaioff et al., 2015). The assumption holds with several common probability distributions including normal, uniform, and (truncated) Laplace, exponential, and logistic distributions.

Remark 1. Note that if the density $f(v)$ is symmetric and the distribution $F(v)$ is log-concave, then $1 - F(v)$ is also log-concave. Moreover, if density $f(v)$ is log-concave, the distribution $F(v)$ is also log-concave. This implies that Assumption 1 is satisfied when density f is symmetric and log-concave.

Remark 2. If a distribution is a Monotone Hazard Rate (MHR) distribution, i.e., $\frac{1-F(v)}{f(v)}$ is decreasing in v, then $1 - F(v)$ is log-concave. This implies that all MHR and symmetric distributions satisfy Assumption 1.

3.2 Objectives

Our goal is to design a dynamic pricing algorithm that (1) optimizes the seller's utility, i.e., the total revenue and (2) keeps individual buyers' valuations private. We use the notion of differential privacy as the measure for the privacy, and use the notion of regret to measure the seller's utility.

Differential privacy. We follow the standard notion of differential privacy and define the privacy notion below.

Definition 1 ((ϵ, δ)-differential privacy (Dwork et al., 2006))**.** A pricing policy $\mathcal{A}$ maps a sequence of preference shocks $\mathbf{Z} = \{z_1, ..., z_T\}$ and an arbitrary (adversarial) sequence of observed product features $\mathbf{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_T\}$ to a sequence of prices $\mathbf{P} = \{p_2, ..., p_T\} \in \mathbb{R}^{T-1}$, i.e., $\mathcal{A}(\mathbf{X}, \mathbf{Z}) = \mathbf{P}$. A randomized pricing

[3]The assumption of zero-mean distribution is without loss of generality. If the mean of the z_t distribution is non-zero, we can add the bias term as the $d+1$-th dimension of θ and make the distribution of z_t to be 0.

policy $\mathcal{A}$ is (ϵ, δ)-differentially private if for any two neighboring preference shock sequences $\mathbf{Z}$ and $\mathbf{Z}'$ that differ in at most one entry, and for all $\mathcal{P} \subset \mathbb{R}^{T-1}$, it holds:

$$\Pr(\mathcal{A}(\mathbf{X}, \mathbf{Z}) \in \mathcal{P}) \leq e^{\epsilon} \Pr(\mathcal{A}(\mathbf{X}, \mathbf{Z}') \in \mathcal{P}) + \delta. \tag{2}$$

If $\delta = 0$, we say that $\mathcal{A}$ is ϵ-differentially private.

Intuitively, the above notion of differential privacy requires that changing any single z_t does not change the probability distribution of the price sequence significantly. Hence, the output of pricing policy $\mathcal{A}$ will only reveal limited information about the buyer no matter he participates or not.

Seller's regret. The seller's utility is measured using regret, which is the maximum expected revenue loss relative to an oracle optimal policy that knows the hidden model parameter $\boldsymbol{\theta}$ in hindsight. Note that the expected revenue from a posted price p is given by:

$$p \cdot \Pr(v_t \geq p) = p(1 - F(p - \boldsymbol{x}_t \cdot \boldsymbol{\theta})).$$

Using the first order condition, we can obtain the optimal price $p^*(\boldsymbol{x}_t)$:

$$p^*(\boldsymbol{x}_t) = \frac{1 - F\big(p^*(\boldsymbol{x}_t) - \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle\big)}{f\big(p^*(\boldsymbol{x}_t) - \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle\big)}. \tag{3}$$

In the following discussion, we use p_t^* to denote $p_t^*(\boldsymbol{x}_t)$ to simplify the presentation. Define the buyer's virtual valuation $\Phi(v) = v - (1 - F(v))/f(v)$ and define the optimal pricing function $\Psi(v) = v + \Phi^{-1}(-v)$. By Assumption 1, Φ is injective and hence Ψ is well-defined. Moreover, it is easy to verify that Ψ is non-negative. Thus, the optimal price can be defined as follows:

$$p_t^* = \Psi(\langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle). \tag{4}$$

We now formally define the regret of a seller's pricing policy. Let $\mathcal{A}$ be the seller's policy that sets price p_t at period t, and p_t can depend on the history of events up to time t. The worst-case regret is defined as:

$$\mathsf{Regret}_{\mathcal{A}}(T) = \sup_{\mathbf{X}, \boldsymbol{\theta}} \sum_{t=1}^{T} \big(p_t^* \mathbb{I}(v_t \geq p_t^*) - p_t \mathbb{I}(v_t \geq p_t)\big). \tag{5}$$

Note that the dependence of $\mathbf{X}, \boldsymbol{\theta}$ are encoded in v_t and p_t.

4 Our Algorithm

In this section we present our private pricing algorithm, which is differentially private and achieves regret of the same order compared to the non-private pricing policy. Our algorithm builds on techniques used in online optimization and a tree-based privacy algorithm for continual observations. We first briefly review the techniques we use.

Private follow the approximate leader. The contextual dynamic pricing problem can be casted as online learning problems with the goal of optimizing certain functions, as will be defined later. Thakurta and Smith (2013) have developed *Private Follow The Approximate Leader (PFTAL)*, a differentially private algorithm for online convex optimization, which takes as input a sequence of convex functions and outputs a sequence of points that minimizes regret. PFTAL is also shown to satisfy the predefined privacy guarantee simultaneously.

PFTAL is adapted from a subgradient descent type algorithm, Follow The Approximate Leader (FTAL) (Hazan et al., 2007), a variant of the Follow The Regularized Leader algorithm. Compared to the standard

Follow The Regularized Leader algorithm, FTAL uses quadratic approximations $\tilde{f}_1, ..., \tilde{f}_T$ to compute the subgradient updates instead of the functions $f_1, ..., f_T$. PFTAL then ensures the privacy guarantee by adopting a tree-based aggregation protocol for releasing the cumulative sums of gradients in a differentially private manner.

Tree-Based Aggregation Protocol. Our notion of privacy is for continual observations, which was first introduced by Dwork et al. (2010). The tree-based aggregation protocol, proposed by Chan et al. (2011), which we refer as TBAP, was designed to ensure a more efficient accuracy-privacy trade-off. The protocol maintains a binary tree whose T leaves correspond to the T entries in the input sequence. Each node in the tree maintains a noisy (privacy-preserving) sum of the input entries in its subtree – by construction, each input affects at most only $\log(T)$ nodes of the tree. The sum at each internal node is $(\epsilon/\log(T))$-differentially private. By the composition property of differential privacy (Dwork et al., 2006), the entire tree is thus ϵ-differentially private. This protocol is the key ingredient of a variety of works that deal with privacy in online settings, including the above mentioned online convex optimization (Jain et al., 2012; Thakurta and Smith, 2013), online submodular optimization (Cardoso and Cummings, 2019) and contextual bandits (Shariff and Sheffet, 2018).

4.1 Our private pricing algorithm

In this section, we present our algorithm which is built on PFTAL. We first note that we cannot directly apply the general framework of online learning algorithms, including FTAL, to our problem. In online learning algorithms, it is assumed that (i) the loss function ℓ_t at every time step is convex, and (ii) the first-order information of the loss functions are available.[4] However, to cast our dynamic pricing problem as an online learning problem, from our regret notion in Equation (5), the loss function ℓ_t is the negative of the revenue obtained in time period t, i.e., $\ell_t = -p_t \mathbb{I}(p_t \geq v_t)$. It is easy to see that the loss functions are not convex. Moreover, the first order information of previous loss functions depend on the corresponding realized valuations $v_1, ..., v_{t-1}$, which are never revealed to the seller.

To address this challenge, below we show that instead of directly formulating the dynamic pricing problem as a online learning problem with non-convex loss functions, we can define an alternative problem with convex loss. Furthermore, as we will prove later in Section 5, the solution for the alternative online learning problem leads to a sublinear regret in the dynamic pricing problem.

An alternative online learning formulation. We now describe the intuitions in formulating the alternative online learning problem. As we can see in Equation (4), computing the optimal price can be reduced to obtaining an accurate estimate of the hidden parameter $\boldsymbol{\theta}$. Moreover, we can update the estimate on $\boldsymbol{\theta}$ through maximum likelihood estimations using only the sale outcomes from the previous rounds. Now consider the online learning problem with loss function being the negative of the log-likelihood function (as a function of $\boldsymbol{\theta}$ instead of v):

$$\ell_t(\boldsymbol{\theta}) = -\mathbb{I}\{y_t = 1\} \log(1 - F(p_t - \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle)) - \mathbb{I}\{y_t = 0\} \log(F(p_t - \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle)),$$

where $y_t \in \{0, 1\}$ indicates whether there is a sale with posted price p_t at time t: $y_t = \mathbb{I}\{p_t \geq v_t\}$. With this definition, by Assumption 1, the loss functions are convex in $\boldsymbol{\theta}$, and the first-order information can also be computed. Therefore, we can apply standard online learning algorithms on this alternative online learning problem.

Note that the solution of this alternative online learning problem does not trivially imply a sublinear regret in the dynamic pricing problem with regret refined in Equation (5). In Section 5, we show how to construct their connections via a series of inequalities.

[4] Or one can query the value of function and then further compute the gradient information.

Differentially private solutions. Below we describe how to solve the alternative online learning problem and how to obtain a differentially private dynamic pricing algorithm from the solutions by borrowing ideas from PFATL. The steps are summarized in Algorithm 1.

We first regularize the loss function to ensure strong convexity. Define the following H-regularized loss function,

$$\ell_t^H(\boldsymbol{\theta}) = \ell_t(\boldsymbol{\theta}) + \frac{H}{2}\|\boldsymbol{\theta}\|^2, \tag{6}$$

where H a parameter that we will tune to optimize the regret bound. Each ℓ_t^H is now H-strongly convex.

The key step of the algorithm is to use the quadratic approximations $\tilde{\ell}_1^H, \ldots, \tilde{\ell}_T^H$ of the loss functions $\ell_1^H, \ldots, \ell_T^H$. At every time step, the algorithm will generate a vector $\widehat{\boldsymbol{\theta}}_t \in \mathbb{R}^d$ as the estimate of $\boldsymbol{\theta}$ which will be further used to compute the price p_t according to Equation (4). Let $\widehat{\boldsymbol{\theta}}_1, \ldots, \widehat{\boldsymbol{\theta}}_t$ be the sequence of estimates up to time t. With strong convexity of ℓ_t^H at hand, we can then lower bound ℓ_t^H on every point in Θ by the following paraboloid:

$$\tilde{\ell}_t^H = \ell_t^H(\widehat{\boldsymbol{\theta}}_t) + \langle \nabla \ell_t^H(\widehat{\boldsymbol{\theta}}_t), \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle + \frac{H}{2}\|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t\|^2. \tag{7}$$

Note that by approximation, $\tilde{\ell}_t^H$ and ℓ_t^H have the same value and gradient at $\widehat{\boldsymbol{\theta}}_t$. We will update $\tilde{\boldsymbol{\theta}}_t$ according to the "leader" of the previous cumulative losses: let $\tilde{\boldsymbol{\theta}}_{t+1} = \arg\min_{\boldsymbol{\theta}\in\Theta} \sum_{\tau=1}^t \tilde{\ell}_\tau^H(\boldsymbol{\theta})$ be the "leader" of previous loss functions $\tilde{\ell}_1^H, \ldots, \tilde{\ell}_t^H$. Ignoring the constant term, we can write $\tilde{\boldsymbol{\theta}}_{t+1}$ as follows:

$$\tilde{\boldsymbol{\theta}}_{t+1} = \arg\min_{\boldsymbol{\theta}\in\Theta} \langle \sum_{\tau=1}^t \nabla \ell_\tau^H(\widehat{\boldsymbol{\theta}}_\tau), \boldsymbol{\theta} \rangle + \frac{H}{2}\sum_{\tau=1}^t \|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_\tau\|^2.$$

Since the optimal pricing function in Equation (4) is injective, to ensure the privacy of p_t w.r.t. z_t, by the immunity of differential privacy to any post-processing computations(Dwork et al., 2014), it suffices to ensure the privacy w.r.t. the estimate $\widehat{\boldsymbol{\theta}}_t$, which is completely determined by the cumulative gradient information $\boldsymbol{\omega}_t = \sum_{\tau=1}^t \nabla \ell_\tau^H(\widehat{\boldsymbol{\theta}}_t)$.[5]

Then the problem is reduced to compute an well-approximated private version $\widehat{\boldsymbol{\omega}}_t$ for $\boldsymbol{\omega}_t$ while still maintaining accuracy. To achieve this, we utilize Tree-Based Aggregation Protocol to compute $\widehat{\boldsymbol{\omega}}_t$. The details of such private aggregation protocol is given in Appendix B.

Armed with the private $\widehat{\boldsymbol{\omega}}_t$, we can compute the private version $\widehat{\boldsymbol{\theta}}_t$ for $\tilde{\boldsymbol{\theta}}_t$ as follows:

$$\widehat{\boldsymbol{\theta}}_{t+1} = \arg\min_{\boldsymbol{\theta}\in\Theta} \langle \sum_{\tau=1}^t \widehat{\boldsymbol{\omega}}_t, \boldsymbol{\theta} \rangle + \frac{H}{2}\sum_{\tau=1}^t \|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_\tau\|^2. \tag{8}$$

The price p_{t+1} will be computed according to optimal price function (4) based on current parameter estimate $\widehat{\boldsymbol{\theta}}_{t+1}$.

5 Analysis on Privacy and Regret

In this section, we present the analysis of the privacy and regret guarantee of Algorithm 1. The informal result of this section is stated as below.

Theorem 1 (Informal)**.** *Consider the valuation function defined in Equation* (1)*. Under Assumption 1, Algorithm 1 is an ϵ-differentially private algorithm that achieves regret in the order of $\tilde{\mathcal{O}}(\sqrt{dT}/\epsilon)$.*

[5]Without additional knowledge about the private data, the computations performed on the output of a differentially private algorithm are also differentially private. Let $\mathcal{A} : \mathcal{D} \to \mathbb{R}$ be (ϵ, δ)-differentially private, and let $f : \mathbb{R} \to \mathbb{R}'$ be an arbitrary randomized function. Then $f \circ \mathcal{A} : \mathcal{D} \to \mathbb{R}'$ is (ϵ, δ)-differentially private.

Algorithm 1 Private Pricing Algorithm

Input: Set $\Theta \subseteq \mathbb{R}^d$, pricing function $\Psi(\cdot)$, privacy constraint ϵ, strong convexity parameter H, Lipschitz parameter u_F.
Input: Product features $\{\boldsymbol{x}_t\}_{t\geq 1}$ (arrivals in an online sequence).
At $t = 1$, select an arbitrary $\widehat{\boldsymbol{\theta}}_1 \in \Theta$, set price $p_1 = 0$.
$\widehat{\boldsymbol{\omega}}_1 \leftarrow \mathsf{TBAP}(u_F, \epsilon, \nabla \ell_1^H(\widehat{\boldsymbol{\theta}}_1))$.
for $t = 1, \ldots$ **do**
  $\widehat{\boldsymbol{\theta}}_{t+1} \leftarrow \arg\min_{\boldsymbol{\theta}\in\Theta} \langle \widehat{\boldsymbol{\omega}}_t, \boldsymbol{\theta}\rangle + \frac{H}{2}\sum_{\tau=1}^{t} \|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_\tau\|^2$.
  Set the price $p_{t+1} \leftarrow \Psi\big(\langle \boldsymbol{x}_{t+1}, \widehat{\boldsymbol{\theta}}_{t+1}\rangle\big)$, observe y_{t+1}.
  $\widehat{\boldsymbol{\omega}}_{t+1} \leftarrow \mathsf{TBAP}(u_F, \epsilon, \nabla \ell_{t+1}^H(\widehat{\boldsymbol{\theta}}_{t+1}))$.
end for

5.1 Privacy guarantee

Recall that a general recipe for achieving privacy is through adding noise, drawn from a particular distribution, to the output of the function. The choice of the noise distribution is scaled with the function's property (i.e., the *sensitivity*). To simplify the presentation, we first define the following quantities:

$$u_F = \sup_{|x|\leq M} \left\{ \max\left\{ -\frac{\partial \log(F(x))}{\partial x}, -\frac{\partial \log(1-F(x))}{\partial x} \right\} \right\},$$
$$w_F = \inf_{|x|\leq M} \left\{ \min\left\{ -\frac{\partial^2 \log(F(x))}{\partial x^2}, -\frac{\partial^2 \log(1-F(x))}{\partial x^2} \right\} \right\},$$

where u_F and w_F characterize the shape of the function $\log F$.[6] Since both F and $1-F$ are log-concave, we have $w_F > 0$. It is easy to see that the loss function $\ell_t(\boldsymbol{\theta})$ is also u_F-Lipschitz. M is defined as $M = 2W + \Phi^{-1}(0)$. With this definition, $M - W$ will be the maximum price the seller could offer. To see this, note that by 1-Lipschitz property of function Ψ, we have $p_t - \Psi(0) = |\Psi(\langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle) - \Psi(0)| \leq |\langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle - 0| = |\langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle| \leq W$. Thus, $p_t \leq W + \Psi(0) = W + \Phi^{-1}(0)$. Let $u_f = \max_{x\in\mathsf{supp}(f)} f(x)$ and $u'_f = \max_{x\in\mathsf{supp}(f)} f'(x)$. We now give following example on how to compute the values of M, u_F and w_F.

Example 1. Consider the case where θ is restricted in an unit ball, i.e, $\|\theta\| \leq 1$, this implies $W = 1$. For an exponential distribution (in the form of $f(x) = \gamma e^{-\gamma x}$) with parameter $\gamma > 0$, one can compute that M will equal to $\gamma + 2$. For a uniform distribution in range $[-6, 6]$, then M will be 5. From the definition, to reason about the values of u_F and w_F, instead of searching among all support of F, we only need to consider the range of $[-M, M]$. Thus, in the case when $\|\theta\| \leq 1$ and F is uniform distributed in range $[-6, 6]$. We can infer that $u_F = 1$ and $w_F = 1$.

Note that via the cumulative gradient $\widehat{\boldsymbol{\omega}}_t$, Algorithm 1 uses entire historical information, including the arrived contexts $\{\boldsymbol{x}_t\}_{t\geq 1}$, the price the seller set $\{p_t\}_{t\geq 1}$, and the sale $\{y_t\}_{t\geq 1}$ up to time t to compute the price p_{t+1}. By ensuring that the gradient is differentially private, and informed by the robustness to post-processing property of differential privacy, we conclude our algorithm is private:

Theorem 2 (Privacy guarantee)**.** *Algorithm 1 , together with using* $\mathsf{TBAP}(u_F, \epsilon, \{\nabla \ell_t^H\}_{t\geq 1})$ *as the subroutine, is* ϵ*-differentially private for any sequence of preference* $z_1, \ldots, z_T$*.*

The proof, which utilizes the structure of binary tree and composition property of differential privacy, is omitted due to space constraint and will be included in the appendix of full paper.

[6] The technical reasons for choosing these definitions will be more clear when we attempt to bound inequality (13) from equation (11) and (12).

Remark 3. As mentioned, the private aggregation protocol only use at most $\lceil \log_2 T \rceil + 1$ noisy terms to compute the estimate $\widehat{\boldsymbol{\omega}}_t$. Thus, the estimation error of $\widehat{\boldsymbol{\omega}}_t$ w.r.t. $\boldsymbol{\omega}_t$ can be bounded at the order of $\mathcal{O}\big(\frac{u_F\sqrt{d}\log^2(T)}{\epsilon}\big)$. To see this, note that according to the line 9 in TBAP, the L_2-norm of noise vector $\gamma \in \mathbb{R}^d$ we add to each node is Gamma distributed with the standard deviation $\mathcal{O}\big(\frac{u_F\sqrt{d}\log(T)}{\epsilon}\big)$. Multiply the maximum number of noisy terms will return the estimation error. Compared to non-differentially private contextual dynamic pricing algorithms, in which the best regret is known to be $\tilde{\mathcal{O}}(\sqrt{T})$ (Kleinberg and Leighton, 2003; Broder and Rusmevichientong, 2012; Javanmard, 2017), the above noise we add to ensure ϵ-differentially private only increase our regret by a factor of $\sqrt{d}/\epsilon$. The analysis of this regret guarantee is described below.

5.2 Regret guarantee

In terms of regret guarantee, we show that Algorithm 1 enjoys the regret of $\tilde{\mathcal{O}}(\sqrt{dT}/\epsilon)$. On a high level, our proof proceeds with the following key steps.

- **Step 1**: Intuitively, to achieve optimal pricing, it requires the algorithm to accurately estimate the unknown parameter $\boldsymbol{\theta}$. Thus, we first show that the Algorithm 1's cumulative prediction error for the parameter $\boldsymbol{\theta}$, defined as follows,

$$\mathsf{Error}(T) = \sum_{t=1}^{T} \langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle^2,$$

  is upper bounded by the "pseudo-regret" incurred on function ℓ_ts.

- **Step 2**: We show how to reduce the problem on bounding $\mathbb{E}[\mathsf{Regret}_{\mathcal{A}}(T)]$ to the problem on bounding the "pseudo-regret" on function ℓ_t through connecting through $\mathsf{Error}(T)$.

- **Step 3**: The third step relates the regret on ℓ_t^H (which our algorithm operates over) to the "pseudo-regret" on ℓ_t, hence establishing the bound on $\mathbb{E}[\mathsf{Regret}_{\mathcal{A}}(T)]$ using the regret on ℓ_t^H.

- **Step 4**: The last step utilizes a generic result of differentially private online learning algorithm to bound the regret incurred on ℓ_t^H.

Step 1 In this step, we show that the cumulative prediction error $\mathsf{Error}(T)$ can be upper bounded by the "pseudo-regret" incurred on function ℓ_ts, namely, $\sum_{t=1}^{T}\big(\ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})\big)$. The analysis will rely on Taylor approximation and other inequalities. The following lemma summarizes the results of this step.

Lemma 1 (Accuracy of $\{\widehat{\boldsymbol{\theta}}_t\}$)**.** *Let $\widehat{\boldsymbol{\theta}}_t$ be the solution of the optimization problem* (8)*, then, under Assumption 1, with probability at least $1 - 1/T^2$, we have*

$$\mathsf{Error}(T) \leq \frac{4}{w_F}\sum_{t=1}^{T}\big(\ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})\big) + \frac{2u_F^2}{w_F^2}\log T. \tag{9}$$

To prove Lemma 1, consider the Taylor expansion of following function $f(x) : f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots$. Then for some point $\boldsymbol{\theta}'$ on the line segment joining $\boldsymbol{\theta}$ and $\widehat{\boldsymbol{\theta}}_t$, we have

$$\ell_t(\boldsymbol{\theta}) - \ell_t(\widehat{\boldsymbol{\theta}}_t) = \langle \nabla\ell_t(\boldsymbol{\theta}_t), \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t\rangle - \frac{1}{2}\langle \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t, \nabla^2\ell_t(\boldsymbol{\theta}')(\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t)\rangle. \tag{10}$$

Recall that we can write the gradient and Hessian of ℓ_t as follows:

$$\nabla\ell_t(\boldsymbol{\theta}) = \tau_t(\boldsymbol{\theta})\boldsymbol{x}_t, \quad \nabla^2\ell_t(\boldsymbol{\theta}) = \chi_t(\boldsymbol{\theta})\boldsymbol{x}_t^\top\boldsymbol{x}_t,$$

where

$$\tau_t(\boldsymbol{\theta}) = \begin{cases} -\frac{\partial \log(F(\kappa_t(\boldsymbol{\theta})))}{\partial x}, & y_t = 0 \\ -\frac{\partial \log(1-F(\kappa_t(\boldsymbol{\theta})))}{\partial x}, & y_t = 1 \end{cases} \tag{11}$$

$$\chi_t(\boldsymbol{\theta}) = \begin{cases} -\frac{\partial^2 \log(F(\kappa_t(\boldsymbol{\theta})))}{\partial x^2}, & y_t = 0 \\ -\frac{\partial^2 \log(1-F(\kappa_t(\boldsymbol{\theta})))}{\partial x^2}, & y_t = 1 \end{cases} \tag{12}$$

And $\kappa_t(\boldsymbol{\theta}) = p_t - \langle \boldsymbol{x}_t, \boldsymbol{\theta} \rangle$. It is easy to check that $|\kappa_t(\boldsymbol{\theta})| \leq M$. Recall that u_F is defined as the upper bound of the first derivative of $\ell_t(\boldsymbol{\theta})$, thus $\tau_t(\boldsymbol{\theta}) \leq u_F$. By the definition of w_F, we have $\chi_t(\boldsymbol{\theta}) \geq w_F$, which further implies that $\nabla^2 \ell_t(\boldsymbol{\theta}') \succeq w_F \boldsymbol{x}_t^\top \boldsymbol{x}_t$. Plugging in Equation (10), we obtain

$$\ell_t(\boldsymbol{\theta}) - \ell_t(\widehat{\boldsymbol{\theta}}_t) \leq \langle \nabla \ell_t(\boldsymbol{\theta}), \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle - \frac{w_F}{2} \langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle^2. \tag{13}$$

We next bound the first term of RHS in above inequality (13). Let Σ_t be the σ-algebra generated by $\{z_t\}_{t \geq 1}$, i.e., Σ_t encode the history information till to time t. Let $J_t = \langle \nabla \ell_t(\boldsymbol{\theta}), \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle = \tau_t(\boldsymbol{\theta}) \langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle$. It is easy to see that we have:

$$\mathbb{E}[J_t | \Sigma_{t-1}] = \mathbb{E}[\tau_t(\boldsymbol{\theta}) | \Sigma_{t-1}] \langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle = 0.$$

Thus, $J(T) = \sum_{t=1}^T J_t$ is a martingale adapted to the filtration $\{\Sigma_t\}_{t \geq 1}$.

Lemma 2. *Consider the above martingale difference sequence $\{J_t\}_{t \geq 1}$ adapted to the filtration $\{\Sigma_t\}$, then with the probability at least $1 - 1/T^2$, we have*

$$J(T) \leq 2u_F \sqrt{\log T} \cdot \mathsf{Error}^{1/2}(T). \tag{14}$$

The proof relies on concentration inequalities to bound the deviations of $J(T)$ from the "high-probability" behavior.

Proof. Recall that $\tau_t(\boldsymbol{\theta})$ is upper bounded by u_F, thus conditioned on Σ_{t-1}, we have $|J_t| \leq \delta_t$ where $\delta_t = u_F |\langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t \rangle|$. Consider the moment generating function $\mathbb{E}[e^{\lambda J_t}]$, for any $\lambda \in \mathbb{R}$, we have

$$\begin{aligned} \mathbb{E}[e^{\lambda J_t} | \Sigma_{t-1}] &\leq \mathbb{E}\left[\frac{\delta_t - J_t}{2\delta_t} e^{-\lambda \delta_t} + \frac{\delta_t + J_t}{2\delta_t} e^{\lambda \delta_t} | \Sigma_{t-1} \right] \\ &\leq \mathbb{E}\left[\frac{e^{-\lambda \delta_t} + e^{\lambda \delta_t}}{2} \right] + \mathbb{E}[J_t | \Sigma_{t-1}] \left(\frac{e^{-\lambda \delta_t} + e^{\lambda \delta_t}}{2\delta_t} \right) \\ &= \cosh(\lambda \delta_t) \leq e^{\lambda^2 \delta_t^2 / 2}, \end{aligned}$$

where the first inequality is due to the convexity of $e^{\lambda x}$. Conditioning on Σ_{t-1} and applying the iterated expectation:

$$\begin{aligned} \mathbb{E}[e^{\lambda J_t}] &= \mathbb{E}\big[e^{\lambda \sum_{t=1}^{T-1} J_t} \mathbb{E}[e^{\lambda J_t} | \Sigma_{T-1}] \big] \\ &\leq \mathbb{E}\big[e^{\lambda \sum_{t=1}^{T-1} J_t} \big] e^{\lambda^2 \delta_T^2 / 2}. \end{aligned}$$

Iterating over time T gives us the following bound:

$$\mathbb{E}[e^{\lambda J(T)}] \leq e^{\lambda^2 \sum_{t=1}^T \delta_t^2 / 2}.$$

By Markov inequality, for any $a \geq 0$, we have

$$\begin{aligned}\Pr(J(T) \geq a) = \Pr(e^{\lambda J(T)} \geq e^{\lambda a}) &\leq e^{-\lambda a}\mathbb{E}[e^{\lambda J(T)}] \\ &\leq e^{-\lambda a + \lambda^2 \sum_{t=1}^T \delta_t^2/2} \\ &\leq e^{-a^2/(2\sum_{t=1}^T \delta_t^2)},\end{aligned}$$

where the last inequality is via optimizing λ.

Applying $a = 2\sqrt{\log T}\big(\sum_{t=1}^T u_F^2 \langle \boldsymbol{x}_t, \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t\rangle^2\big)^{1/2}$ will complete the proof. □

Taking the summation over T and substituting above inequality into (13), we have the following

$$\mathsf{Error}(T) \leq \frac{2}{w_F}\sum_{t=1}^T \big(\ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})\big) + \frac{2u_F}{w_F}\sqrt{\log T}\mathsf{Error}^{1/2}(T)$$

with probability at least $1 - 1/T^2$. Rearranging the terms in the above inequality, we conclude the desired result in Lemma 1.

Step 2 With the above lemma at hand, we show that we can bridge the incurred regret $\mathsf{Regret}_{\mathcal{A}}(T)$ with the cumulative loss $\sum_{t=1}^T \ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})$. This is summarized in the following Lemma.

Lemma 3. *Consider the parametric valuation model defined in Equation* (1)*. Under Assumption 1, the regret of our private pricing policy, Algorithm 1, is bounded as:* $\mathsf{Regret}_{\mathcal{A}}(T) \leq C\bigg(\mathbb{E}\big[\sum_{t=1}^T \big(\ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})\big)\big] + \frac{2u_F^2}{w_F^2}\log T\bigg) + M/T$*, where* $C = \frac{4u_f + 2Mu_f'}{w_F}$.

The proof constructs an inequality between $\mathsf{Regret}_{\mathcal{A}}(T)$ and the prediction error $\mathsf{Error}(T)$. By the result we obtained in Step 1, we can then achieve the above lemma.

Proof. Let $r_t = p_t^*\mathbb{I}(v_t \geq p_t^*) - p_t\mathbb{I}(v_t \geq p_t)$ be the instantaneous regret at time t. Then we have:

$$\begin{aligned}\mathbb{E}[r_t|\Sigma_{t-1}] &= \mathbb{E}[p_t^*\mathbb{I}(v_t \geq p_t^*) - p_t\mathbb{I}(v_t \geq p_t)|\Sigma_{t-1}] \\ &= p_t^*\big(1 - F\big(p_t^* - \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle\big)\big) - p_t\big(1 - F\big(p_t - \langle \boldsymbol{x}_t, \widehat{\boldsymbol{\theta}}_t\rangle\big)\big).\end{aligned}$$

Define function $g(\cdot; u) : \mathbb{R} \to \mathbb{R}$ such that $g(p; u) = p(1 - F(p - u))$. The first-order condition of $g(p; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)$ at p_t^* implies that $g'(p_t^*; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle) = 0$. Take the Taylor expansion of $g(p; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)$ at point p_t^*, then there must exist p' such that:

$$g(p_t; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle) = g(p_t^*; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle) + \frac{1}{2}g''(p'; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)(p_t - p_t^*)^2.$$

The absolute value of $g''(p'; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)$ is upper bounded by a constant $C = 2u_f + Mu_f'$:

$$\begin{aligned}|g''(p'; \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)| &\leq |2f(p' - \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle) + p'f'(p' - \langle \boldsymbol{x}_t, \boldsymbol{\theta}\rangle)| \\ &\leq 2u_f + Mu_f'.\end{aligned}$$

Thus,

$$\begin{aligned}\mathbb{E}[r_t|\Sigma_{t-1}] &= g(p_t;\langle \boldsymbol{x}_t,\boldsymbol{\theta}\rangle) - g(p_t^*;\langle \boldsymbol{x}_t,\boldsymbol{\theta}\rangle)\\ &\le \frac{2u_f + Mu_f'}{2}(p_t - p_t^*)^2\\ &= \frac{2u_f + Mu_f'}{2}\left(\Psi\big(\langle \boldsymbol{x}_t,\boldsymbol{\theta}\rangle\big) - \Psi\big(\langle \boldsymbol{x}_t,\widehat{\boldsymbol{\theta}}_t\rangle\big)\right)^2\\ &\le \frac{2u_f + Mu_f'}{2}\langle \boldsymbol{x}_t,\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t\rangle^2,\end{aligned}$$

where the last inequality is due to the 1-Lipschitz property of function Ψ. Define the event $\mathcal{G}$ when (9) holds. Then the probability of the complement of event $\mathcal{G}$, denoted by $\mathcal{G}^c$, is given by $\Pr(\mathcal{G}^c) \le 1/T^2$. By the law of total expectation, we can now step to upper bound the regret by expected cumulative prediction error:

$$\begin{aligned}\mathsf{Regret}_{\mathcal{A}}(T) &\le \sum_{t=1}^{T}\mathbb{E}[r_t] = \sum_{t=1}^{T}\mathbb{E}[\mathbb{E}[r_t|\Sigma_{t-1}]]\\ &= \sum_{t=1}^{T}\mathbb{E}[\mathbb{E}[r_t\cdot\mathbb{I}(\mathcal{G})|\Sigma_{t-1}] + \mathbb{E}[r_t\cdot\mathbb{I}(\mathcal{G}^c)|\Sigma_{t-1}]]\\ &\le \frac{2u_f + Mu_f'}{2}\sum_{t=1}^{T}\mathbb{E}\big[\langle \boldsymbol{x}_t,\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t\rangle^2\big] + \frac{M}{T}.\end{aligned}$$

With the fact we showed in Lemma 1, the proof is complete. □

Step 3 Now the problem reduces to upper bound the cumulative loss $\sum_{t=1}^{T}\ell_t(\widehat{\boldsymbol{\theta}}_t) - \ell_t(\boldsymbol{\theta})$, which we will bound using the "real" regret incurred on the loss functions $\{\ell_t^H\}_{t\ge 1}$. Note that for any sequence of vectors $\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_T \in \Theta$, we have the following:

$$\sum_{t=1}^{T}\ell_t(\boldsymbol{\theta}_t) - \min_{\boldsymbol{\theta}\in\Theta}\sum_{t=1}^{T}\ell_t(\boldsymbol{\theta}) \le \sum_{t=1}^{T}\ell_t^H(\boldsymbol{\theta}_t) - \min_{\boldsymbol{\theta}\in\Theta}\sum_{t=1}^{T}\ell_t^H(\boldsymbol{\theta}) + \frac{HT}{2}W^2.$$

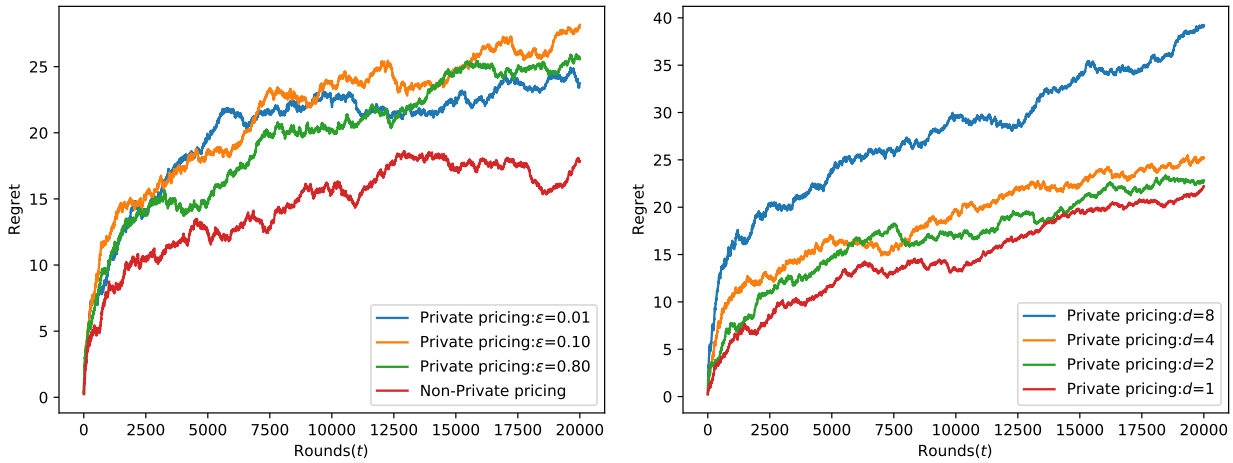
This is due to the approximation we defined in Equation (7).

Step 4 We are now ready to prove an upper bound on the regret of Algorithm 1 when the prices are set to satisfy the desired privacy guarantee. The regret bound of Algorithm 1 is given in following theorem:

Theorem 3 (Regret guarantee)**.** *Consider the value function defined in Equation* (1)*. Under the Assumption 1, for any sequence of arriving product contexts, the expected regret of Algorithm 1 is upper bounded by* $\mathbb{E}[\mathsf{Regret}_{\mathcal{A}}(T)] = \mathcal{O}\left(\frac{C\sqrt{\log^{2.5}T}\left(u_F + 2W\sqrt{\frac{d\log^{2.5}T}{\epsilon T}}\right)^2}{\epsilon}\sqrt{T}\right) + C'\log T + M/T$*, where* $C = \frac{4u_f + 2Mu_f'}{w_F}\sqrt{d}$ *and* $C' = \frac{2Cu_F^2}{w_F^2\sqrt{d}}$.

To prove this theorem, we reuse the analysis by Thakurta and Smith (2013) via the following lemma:

Lemma 4. *Let* $f_1,\ldots,f_T$ *be L-Lipschitz, H-strongly convex functions and* $\mathcal{C}\in\mathbb{R}^d$ *be the compact convex set. Then the expected regret of PFTAL satisfies* $\mathbb{E}[\sum_{t=1}^{T}f_t(\boldsymbol{\theta}_t) - \min_{\boldsymbol{\theta}\in\mathcal{C}}\sum_{t=1}^{T}f_t(\boldsymbol{\theta})] = \mathcal{O}\left(\frac{d(L+H\|\mathcal{C}\|)^2\log^{2.5}(T)}{\epsilon H}\right)$.



(a) Regret over time: varying ϵ for Algorithm 1.

(b) Regret over time: varying d for Algorithm 1.

Figure 1: Examining the algorithm performance with different ϵ and different d.

Proof. Note that in Equation (6), we add the L_2-regularizer to ℓ_t. The loss function ℓ_t^H is H-strongly convex, and it has the following Lipschitz property:

$$\begin{aligned}|\ell_t^H(\boldsymbol{\theta}_1) - \ell_t^H(\boldsymbol{\theta}_2)| &\leq |\ell_t(\boldsymbol{\theta}_1) - \ell_t(\boldsymbol{\theta}_2)| + \frac{H}{2}\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|^2 \\ &\leq u_F\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\| + \frac{H}{2}\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|\|\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2\| \\ &\leq (u_F + HW)\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|,\end{aligned}$$

where the last equality is due to $\|\boldsymbol{\theta}\| \leq W$ for all $\boldsymbol{\theta} \in \Theta$. Thus, invoking the result in Lemma 4 will yield us $\mathbb{E}\big[\sum_{t=1}^T \ell_t(\widehat{\boldsymbol{\theta}}_t) - \min_{\boldsymbol{\theta}\in\Theta}\sum_{t=1}^T \ell_t(\boldsymbol{\theta})\big] = \mathcal{O}\big(\frac{d(u_F+2HW)^2\log^{2.5}(T)}{\epsilon H}\big) + \frac{HT}{2}W^2$. Together with the results we show in Step 3, above regret bound can be achieved by setting $H = \mathcal{O}\left(\sqrt{\frac{d\log^{2.5}(T)}{\epsilon T}}\right)$. □

Remark 4. Note that this regret bound is robust to adaptively arrived adversarial product contexts $\{\boldsymbol{x}_t\}_{t\geq 1}$. By simplifying the constants and ignoring the logarithmic terms, the above regret bound is reduced to the bound of the order $\tilde{\mathcal{O}}(\frac{\sqrt{dT}}{\epsilon})$, which is worse than the non-private regret bound of $\tilde{\mathcal{O}}(\sqrt{T})$, up to a constant factor $\frac{\sqrt{d}}{\epsilon}$. This is due to the noise we added to the updates $\tilde{\boldsymbol{\theta}}_t$.

6 Simulations

In this section, we demonstrate the simulation results which characterize the performance of Algorithm 1.

We first describe the common setting. Given a dimension $d = 4$, we set the unknown parameter $\boldsymbol{\theta} = \frac{\boldsymbol{z}}{\|\boldsymbol{z}\|}$, where $\boldsymbol{z}$ is draw from a multivariate Gaussian distribution, $\boldsymbol{z} \sim \mathsf{N}(0, \mathrm{I}_d)$. At each time step, the product features $\boldsymbol{x}_t$ are chosen independently from a $\mathsf{N}(0, \mathrm{I}_d)$ with the L_2-norm normalized to be 1. The preference shock z_t is generated as $z_t \sim \mathsf{N}(0, 1)$.

Our regret analysis indicates the upper bound of the regret of our algorithm is in the order of $\mathcal{O}(1/\epsilon)$. We conduct the simulation to demonstrate the practical performance of our algorithm. We measure the performance of Algorithm 1 via the cumulative pseudo-regret over 2×10^4 rounds.

Examining ϵ We choose several levels of privacy guarantee ($\epsilon = 0.01, 0.1, 0.80$) that Algorithm 1 has to be satisfied. Their performances are compared with the performance of non-private algorithm, which we use FTAL (Hazan et al., 2007).

The results of the simulation are presented in Fig. 1a, which compares the cumulative regret (averaged over 40 trials) over different privacy guarantee, and also the regret of non-private pricing algorithm. As expected, Algorithm 1's regret is larger than the regret of non-private algorithm. However, the performance difference is not as large as suggested by the theoretical regret bound. The possible explanation is that both the regret bounds and the differential privacy are defined as the worst-case notion. The results suggest the algorithm would likely perform much more robustly to different predefined privacy guarantee in practice.

Examining d Our regret bound also indicates a dependency with the context dimension d, i.e., when d increases, the regret will be increasing. Fig. 1b shows the results with total accumulated regret plotted against different dimensions d where the privacy parameter is set to be 0.01, and all the other parameters are the same as the previous experiment. The results validate our theoretical regret dependency and indicate that when the context dimension d becomes larger, the actual performance of Algorithm 1 will degrade.

7 Discussion: Nonlinear Valuation

Our discussion so far focuses on linear valuation models given in Equation (1). While it is a standard model in dynamic pricing, it is natural to ask whether our private pricing framework applies to more general *nonlinear* valuation models. In this section, we explore another set of valuation models in the following form (Javanmard and Nazerzadeh, 2019; Golrezaei et al., 2019):

$$v_t(\boldsymbol{x}_t) = \psi(\langle \phi(\boldsymbol{x}_t), \boldsymbol{\theta} \rangle + z_t), \tag{15}$$

where $\phi : \mathbb{R}^d \to \mathbb{R}^d$ is a product feature mapping function and $\psi : \mathbb{R} \to \mathbb{R}$ is a general strictly increasing and log-concave function.

Remark 5. Note that the above family of nonlinear functions captures many important scenarios such as: (i) Log-log model ($\psi(x) = e^x, \phi = \ln(x)$); (ii) Semi-log model ($\psi(x) = e^x, \phi = x$); and (iii) Logistic model ($\psi(x) = e^x/(1+e^x), \phi = x$), among others.

Below we demonstrate that adopting this family of nonlinear valuation functions in contextual dynamic pricing shares the same order of regret bound as with the linear valuation function under the same privacy guarantee. The only difference in the analysis (when the valuation functions are different) is on how to compute the prices with the computed private estimate $\widehat{\boldsymbol{\theta}}_t$ according to the optimal pricing function as we see in Equation (4) for linear valuation functions.

In particular, since ϕ is strictly increasing, one can compute the optimal price $p_t^*(\boldsymbol{x}_t)$ by the first-order condition:

$$\psi'(\psi^{-1}(p_t^*)) = \frac{p_t^* f(\psi^{-1}(p_t^*) - \langle \phi(\boldsymbol{x}_t), \boldsymbol{\theta} \rangle)}{F(\psi^{-1}(p_t^*) - \langle \phi(\boldsymbol{x}_t), \boldsymbol{\theta} \rangle)}.$$

Let $h(v) = f(v)/(1 - F(v))$ denote the hazard rate function for distribution F and $\tilde{p}_t^* = \psi^{-1}(p_t^*)$. Thus, we can write above equation in the following form:

$$\langle \phi(\boldsymbol{x}_t), \boldsymbol{\theta} \rangle = \tilde{p}_t^* - h^{-1}\left(\frac{\psi'(\tilde{p}_t^*)}{\psi(\tilde{p}_t^*)}\right).$$

Due to log-concavity of function ψ, $\frac{\psi'(v)}{\psi(v)} = \frac{d}{dv}\log\psi(v)$ is monotonically decreasing. Since $1-F$ is also log-concave, then h is increasing, which further implies that $-h^{-1}\big(\psi'(v)/\psi(v)\big)$ is increasing. Thus, function Ψ is strictly increasing and well-defined. Furthermore, it is easy to see that function Ψ is also 1-Lipschitz. Also define

$$\Psi^{-1}(v) = v - h^{-1}\left(\frac{\psi'(v)}{\psi(v)}\right).$$

The optimal price can then be computed as follows:

$$p_t^* = \psi(\Psi(\langle\phi(\boldsymbol{x}_t), \boldsymbol{\theta}\rangle)). \tag{16}$$

The loss function we optimize to update the private estimates $\{\widehat{\boldsymbol{\theta}}_t\}$ is then given by:

$$\ell_t(\boldsymbol{\theta}) = -\mathbb{I}\{y_t = 1\}\log(1 - F(\psi^{-1}(p_t) - \langle\phi(\boldsymbol{x}_t), \boldsymbol{\theta}\rangle)) - \mathbb{I}\{y_t = 0\}\log(F(\psi^{-1}(p_t) - \langle\phi(\boldsymbol{x}_t), \boldsymbol{\theta}\rangle)).$$

Accordingly, in private Algorithm 1, we will compute the current price as $p_t = \psi^{-1}\big(\Psi(\langle\phi(\boldsymbol{x}_t), \widehat{\boldsymbol{\theta}}_t\rangle)\big)$. Moreover, the difference between the posted price and the optimal price can be bounded as $|p_t - p_t^*| = \big|\psi^{-1}\big(\Psi(\langle\phi(\boldsymbol{x}_t), \widehat{\boldsymbol{\theta}}_t\rangle)\big) - \psi^{-1}\big(\Psi(\langle\phi(\boldsymbol{x}_t), \boldsymbol{\theta}\rangle)\big)\big| \leq L_\psi\|\widehat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}\|$, with the Lipschitz constant L_ψ for function ψ. With above modifications, our analysis of Algorithm 1 for linear valuation model extends over to the nonlinear model (15). The order of regret bound we achieve for linear model (1) thus holds for above nonlinear model up to a constant factor.

8 Conclusion and Future work

We explore the design of differentially private algorithms for the contextual dynamic pricing problem. We present an algorithm that is ϵ-*differentially private* and achieves expected regret $\tilde{\mathcal{O}}\big(\frac{\sqrt{dT}}{\epsilon}\big)$, where d is the dimension of product features and T is the time horizon.

Future work include the exploration of more general user valuation models. For example, our current model implicitly assumes the hidden parameter $\boldsymbol{\theta}$ is static over time. In practice, since buyer population might (slowly) evolve, it would be interesting to generalize this model to the case on varying $\{\boldsymbol{\theta}_t\}$ and explore the utility-privacy trade-off in this time-varying setting. Furthermore, since differential privacy only provides a theoretical upper bound on the privacy leak, it would be interesting to explore the practical amount of privacy leak, possibly constrained to a certain set of adversarial strategies in learning buyer valuations from prices.

References

Kareem Amin, Afshin Rostamizadeh, and Umar Syed. 2014. Repeated contextual auctions with strategic buyers. In *Advances in Neural Information Processing Systems*. 622–630.

Moshe Babaioff, Shaddin Dughmi, Robert Kleinberg, and Aleksandrs Slivkins. 2015. Dynamic pricing with limited supply. *ACM Transactions on Economics and Computation (TEAC)* 3, 1 (2015), 4.

Mark Bagnoli and Ted Bergstrom. 2005. Log-concave probability and its applications. *Economic theory* 26, 2 (2005), 445–469.

Maria-Florina Balcan, Avrim Blum, and Yishay Mansour. 2008. Item pricing for revenue maximization. In *Proceedings of the 9th ACM conference on Electronic commerce*. 50–59.

Hamsa Bastani and Mohsen Bayati. 2015. Online decision-making with high-dimensional covariates. *Available at SSRN 2661896* (2015).

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

Josef Broder and Paat Rusmevichientong. 2012. Dynamic pricing under a general parametric choice model. *Operations Research* 60, 4 (2012), 965–980.

Adrian Rivera Cardoso and Rachel Cummings. 2019. Differentially Private Online Submodular Minimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*. 1650–1658.

T-H Hubert Chan, Elaine Shi, and Dawn Song. 2011. Private and continual release of statistics. *ACM Transactions on Information and System Security (TISSEC)* 14, 3 (2011), 26.

Le Chen, Alan Mislove, and Christo Wilson. 2016. An empirical analysis of algorithmic pricing on amazon marketplace. In *Proceedings of the 25th International Conference on World Wide Web*. 1339–1349.

Maxime Cohen, Ilan Lobel, and Renato Paes Leme. 2016. Feature-based dynamic pricing. *ACM Conference on Economics and Computation* (2016).

Arnoud V den Boer and Bert Zwart. 2013. Simultaneously learning and optimizing using controlled variance pricing. *Management science* 60, 3 (2013), 770–783.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. 265–284.

Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N Rothblum. 2010. Differential privacy under continual observation. In *Proceedings of the forty-second ACM symposium on Theory of computing*. 715–724.

Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

Cynthia Dwork, Guy N Rothblum, and Salil Vadhan. 2010. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. 51–60.

Negin Golrezaei, Adel Javanmard, and Vahab Mirrokni. 2019. Dynamic incentive-aware learning: Robust pricing in contextual auctions. In *Advances in Neural Information Processing Systems*. 9756–9766.

Elad Hazan, Amit Agarwal, and Satyen Kale. 2007. Logarithmic regret algorithms for online convex optimization. *Machine Learning* 69, 2-3 (2007), 169–192.

Prateek Jain, Pravesh Kothari, and Abhradeep Thakurta. 2012. Differentially private online learning. In *Conference on Learning Theory*. 24–1.

Adel Javanmard. 2017. Perishability of data: dynamic pricing under varying-coefficient models. *The Journal of Machine Learning Research* 18, 1 (2017), 1714–1744.

Adel Javanmard and Hamid Nazerzadeh. 2019. Dynamic Pricing in High-dimensions. *Journal of Machine Learning Research* 20, 9 (2019), 1–49.

Robert Kleinberg and Tom Leighton. 2003. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.* 594–605.

Ilan Lobel, Renato Paes Leme, and Adrian Vladu. 2018. Multidimensional binary search for contextual decision-making. *Operations Research* 66, 5 (2018), 1346–1361.

Sheng Qiang and Mohsen Bayati. 2016. Dynamic pricing with demand covariates. *Available at SSRN 2765257* (2016).

Roshan Shariff and Or Sheffet. 2018. Differentially private contextual linear bandits. In *Advances in Neural Information Processing Systems*. 4296–4306.

Abhradeep Guha Thakurta and Adam Smith. 2013. (Nearly) optimal algorithms for private online learning in full-information and bandit settings. In *Advances in Neural Information Processing Systems*. 2733–2741.

Francesco Trovò, Stefano Paladino, Marcello Restelli, and Nicola Gatti. 2018. Improving multi-armed bandit algorithms in online pricing settings. *International Journal of Approximate Reasoning* 98 (2018), 196–235.

A Properties of Optimal Pricing Function

Lemma 5 (Javanmard and Nazerzadeh (2019)). *If $1-F$ is log-concave, then the virtual valuation function Φ is strictly monotone increasing and the optimal price function Ψ satisfies $0 < \Psi'(v) < 1$, for all values of $v \in \mathbb{R}$.*

Proof. The virtual valuation function can be writen as $\Phi(v) = v - 1/\lambda(v)$ where $\lambda(v) = \frac{f(v)}{1-F(v)} = -\log'(1 - F(v))$ is the hazard rate function. Since $1-F$ is log-concave, the hazard function $\lambda(v)$ is increasing which implies that $\Phi(v)$ is strictly increasing. Indeed, by this argument $\Phi'(v) > 1$. Recalling the definition $\Psi(v) = v + \Phi^{-1}(-v)$, we have $\Psi'(v) = 1 - \frac{1}{\Phi'(\Phi^{-1}(-v))}$. Since Φ is strictly increasing, we have $\Psi'(v) < 1$. $\Psi'(v) > 0$ follows that we have $\Phi'(\Phi^{-1}(-v)) > 1$ for all $v \in \mathbb{R}$. □

B The Tree-Based Aggregation Protocol

We consider the problem of computing partial sums while preserving differential privacy. Formally, let $\mathbf{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_T\}$ be a sequence of vectors, where at time t, a new vector $\boldsymbol{v}_t \in \mathbb{R}^d$ arrives. The goal is to output $\boldsymbol{s}_t$, which is a privacy-preserved version of cumulative sum $\sum_{\tau=1}^{t} \boldsymbol{v}_\tau$ at each time t, without compromising too much of the accuracy. We use Tree-Based Aggregation Protocol (Chan et al., 2011; Dwork et al., 2010) to ensure the privacy guarantee. Let Γ be a complete binary tree with its leaf nodes being $l_1, ..., l_T$. Each internal node stores the sum of all the leaf nodes in the sub-tree rooted at the node. First note that one can compute any partial sum $\boldsymbol{s}_t$ using at most $\lceil \log_2 T \rceil + 1$ nodes of Γ. Thus, if the sum at each internal node is $(\epsilon/\log_2(T))$-differentially private, by the composition property of differential privacy (Dwork et al., 2010), the entire tree is ϵ-differentially private.

Algorithm 2 Private Tree Based Aggregation Protocol

Input: A sequence of vector $(\boldsymbol{v}_1, ..., \boldsymbol{v}_T \in \mathbb{R}^d)$ (arrive in sequentially). μ: L_2-norm bound on $\boldsymbol{v}_t$. Privacy guarantee parameter ϵ.
Output: Sequence of noisy partial sums $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_T \in \mathbb{R}$
Initialization: Initialize a binary tree Γ of size $2^{\lceil \log_2 T \rceil + 1} - 1$ with leaves $\boldsymbol{v}_1, ..., \boldsymbol{v}_T$
for $t = 1, \ldots, T$ **do**
  Accept $\boldsymbol{v}_t$ sequentially. Let $L_{\boldsymbol{v}_t \to \mathsf{root}} = \{\boldsymbol{v}_t \to \cdots \to \mathsf{root}\}$ be the path from $\boldsymbol{v}_t$ to the root.
  Let Λ be the first node in $L_{\boldsymbol{v}_t \to \mathsf{root}}$ that is left-child in Γ. Let $L_{\boldsymbol{v}_t \to \Lambda} = \{\boldsymbol{v}_t \to \cdots \to \Lambda\}$.
  for all nodes α in path $L_{\boldsymbol{v}_t \to \mathsf{root}}$ **do**
    $\alpha \leftarrow \alpha + \boldsymbol{v}_t$
    if $\alpha \in L_{\boldsymbol{v}_t \to \Lambda}$, **then** $\alpha \leftarrow \alpha + \gamma$ where $\gamma \in \mathbb{R}^d$ is sampled by $\Pr[\gamma = \widehat{\gamma}] \propto \exp(-\frac{\|\widehat{\gamma}\|_2 \epsilon}{\mu(\lceil \log_2 T \rceil + 1)})$
  end for
  Initialize vector $\boldsymbol{s}_t \in \mathbb{R}^d$ to zero. Let b be a $(\lceil \log_2 T \rceil + 1)$-bit binary representation of t.
  for $i = 1, \ldots, [\log_2 T + 1]$ **do**
    if bit $b_i = 1$ **then**
      if i-th node in $L_{\boldsymbol{v}_t \to \mathsf{root}}$ (denoted $L^i_{\boldsymbol{v}_t \to \mathsf{root}}$) is the left child in Γ, **then** $\boldsymbol{s}_t \leftarrow \boldsymbol{s}_t + L^i_{\boldsymbol{v}_t \to \mathsf{root}}$
      else $\boldsymbol{s}_t \leftarrow \boldsymbol{s}_t$+left sibling of $L^i_{\boldsymbol{v}_t \to \mathsf{root}}$
    end if
  end for
  return noisy partial sum $\boldsymbol{s}_t$
end for

B.1 Proof of Privacy Guarantee

Proof. Note that by the differential privacy's immunity to any post-processing, given $\{\widehat{\boldsymbol{\omega}}_t\}_{t\geq 2}$ (where $\widehat{\boldsymbol{\omega}}_t$ is the noisy version of $\boldsymbol{v}_t = \sum_{\tau=1}^{t} \nabla \ell_\tau^H(\widehat{\boldsymbol{\theta}}_\tau)$), the estimate of hidden parameter $\{\widehat{\boldsymbol{\theta}}_t\}_{t\geq 2}$ would be determined, thus the price $\{p_t\}_{t\geq 2}$ is also determined. Hence, to argue that the Algorithm 1 is differentially private, the problem reduces to argue that the sequence $\{\widehat{\boldsymbol{\omega}}_t\}_{t\geq 2}$ is ϵ-differentially private. Recall that $\mathbf{Z}$ and $\mathbf{Z}'$ are any two sequences which differing in exactly one preference shock. For any set $\mathbf{S} = (\boldsymbol{s}_1, ..., \boldsymbol{s}_T) \subset \mathbb{R}^T$ of T sequence vector sums and let $\mathcal{A}$ denote the mapping inherent in Algorithm 1 and the Tree Aggregation Protocol 2 from $\mathbf{Z}$ to $\mathbf{S}$, then we need to argue that following holds true:

$$\frac{\Pr(\mathcal{A}(\mathbf{X}, \mathbf{Z})) = \mathbf{S}}{\Pr(\mathcal{A}(\mathbf{X}, \mathbf{Z}')) = \mathbf{S}} \prod_{t=2}^{T} \frac{\Pr(\widehat{\boldsymbol{\omega}}_t(\mathbf{Z}) = \boldsymbol{s}_t | \widehat{\boldsymbol{\omega}}_2 = \boldsymbol{s}_2, ..., \widehat{\boldsymbol{\omega}}_{t-1} = \boldsymbol{s}_{t-1})}{\Pr(\widehat{\boldsymbol{\omega}}_t(\mathbf{Z}') = \boldsymbol{s}_t | \widehat{\boldsymbol{\omega}}_2 = \boldsymbol{s}_2, ..., \widehat{\boldsymbol{\omega}}_{t-1} = \boldsymbol{s}_{t-1})} \leq e^{\epsilon}. \tag{17}$$

Since the computation of $\{\widehat{\boldsymbol{\omega}}_t\}_{t\geq 2}$ is via the tree Γ, the output of $\mathcal{A}$ is determined by the nodes of Γ. Let $\Gamma(\mathbf{Z}) = (\alpha_1(\mathbf{Z}), ..., \alpha_{2^{\lceil \log_2 T \rceil + 1} - 1}(\mathbf{Z}))$ denote the in-order tree traversal of $\Gamma(\mathbf{Z})$ and the value stored in the node $\alpha_t(\mathbf{Z})$ is denoted by α_t. Thus, the proof of inequality (17) reduces to prove following:

$$\frac{\Pr(\Gamma(\mathbf{Z})) = \mathbf{S}}{\Pr(\Gamma(\mathbf{Z}')) = \mathbf{S}} \prod_{t=2}^{n} \frac{\Pr(\alpha_t(\mathbf{Z}) = \alpha_t | \alpha_1(\mathbf{Z}) = \alpha_1, ..., \alpha_t(\mathbf{Z}) = \alpha_{t-1})}{\Pr(\alpha_t(\mathbf{Z}') = \alpha_t | \alpha_1(\mathbf{Z}') = \alpha_1, ..., \alpha_t(\mathbf{Z}') = \alpha_{t-1})} \leq e^{\epsilon}. \tag{18}$$

where $n = 2^{\lceil \log_2 T \rceil + 1} - 1$. As mentioned, changing any single entry in $\mathbf{Z}$ have only limited affects on $\lceil \log_2 T \rceil + 1$ terms in (18). By the amount of noise added to each node of the tree (where the noise is carefully computed according to Lipschitz property of $\ell_t^H(\boldsymbol{\theta})$), each of the ratio in the product of (18) is bounded by $\exp(\frac{\epsilon}{\lceil \log_2 T \rceil + 1})$. Thus, we can conclude that the Algorithm 1 is ϵ-differentially private. □