A Profit-Maximizing Data Marketplace with Differentially Private Federated Learning under Price Competition

PENG SUN, Hunan University, China
LIANTAO WU*, East China Normal University, China
ZHIBO WANG, Zhejiang University, China
JINFEI LIU, Zhejiang University, China
JUAN LUO, Hunan University, China
WENQIANG JIN, Hunan University, China

The proliferation of machine learning (ML) applications has given rise to a new and popular data marketplace paradigm. These marketplaces facilitate ML model requesters in obtaining data from data owners to train their desired models. To mitigate the privacy concerns of data owners, federated learning (FL) has been introduced, enabling collaborative model training without raw data trading. Furthermore, researchers have incorporated differential privacy (DP) techniques into FL, resulting in differentially private federated learning (DPFL) to enhance privacy preservation. However, existing designs of DPFL-based data marketplaces consider a *simplified but unrealistic* scenario where the model requester holds dominant market power, and data owners cannot set their own prices. In this work, we propose a novel DPFL-based data marketplace that accommodates both price-taking and price-setting data owners. We model the interactions among the model requester and these two types of data owners as a three-stage Stackelberg game, focusing on maximizing the model requester's profit. We rigorously establish that the formulated game is a convex game with a unique subgame perfect equilibrium. Moreover, we devise iterative algorithms to determine the equilibrium strategies for the model requester and price-setting data owners. Notably, our algorithms allow data owners to operate without requiring complete information about the model requester or other data owners. Numerical experiments demonstrate the superiority of our proposed three-stage framework in terms of the model requester's profitability compared to scenarios where only price-taking data owners are involved. Furthermore, we reveal that price competition among price-setting data owners reduces equilibrium market prices.

CCS Concepts: • **Security and privacy**; • **Economics of security and privacy**;

Additional Key Words and Phrases: Data marketplace, federated learning, differential privacy, Stackelberg game, subgame perfect equilibrium

ACM Reference Format:
Peng Sun, Liantao Wu, Zhibo Wang, Jinfei Liu, Juan Luo, and Wenqiang Jin. 2024. A Profit-Maximizing Data Marketplace with Differentially Private Federated Learning under Price Competition. *Proc. ACM Manag. Data* 2, 4 (SIGMOD), Article 191 (September 2024), 27 pages. https://doi.org/10.1145/3677127

*Liantao Wu is the corresponding author.

Authors' Contact Information: Peng Sun, psun@hnu.edu.cn, Hunan University, China; Liantao Wu, ltwu@sei.ecnu.edu.cn, East China Normal University, China; Zhibo Wang, zhibowang@zju.edu.cn, Zhejiang University, China; Jinfei Liu, jinfeiliu@zju.edu.cn, Zhejiang University, China; Juan Luo, juanluo@hnu.edu.cn, Hunan University, China; Wenqiang Jin, wqjin@hnu.edu.cn, Hunan University, China.

ACM 2836-6573/2024/9-ART191
https://doi.org/10.1145/3677127

1 Introduction

1.1 Background

Recent years have witnessed an unprecedented prevalence of machine learning (ML) applications (e.g., face recognition [16]). However, a major challenge in developing high-usability ML models lies in acquiring a substantial amount of high-quality training data. This hurdle has greatly facilitated the commoditization of data and led to the emergence of diverse ML-oriented data marketplaces (e.g., Innodata [14], DEFINED.AI [7], and Zaloni [48]). These data marketplaces offer ML model requesters, such as facial recognition service providers, the opportunity to procure training data (e.g., facial images) from data owners (e.g., mobile users), enabling them to train their desired models (e.g., a convolutional neural network (CNN)-based facial recognition model) [28]. Meanwhile, the relevant data pricing accounts for the costs associated with data collection and data management of data owners.

In existing escrow-based data marketplaces (e.g., [1, 5, 6, 20]), data owners need to share their raw training data with a third party, which may be *honest but curious* or even hacked, leading to privacy breaches. With data owners' increased privacy awareness and the implementation of stringent privacy regulations like the General Data Protection Regulation (GDPR), the centralization of raw data has become increasingly challenging or even infeasible. Therefore, such data marketplaces have become impractical. Federated learning (FL) [27, 34, 51] is a privacy-preserving distributed ML paradigm that decouples model training from the need to centralize training data through distributed learning and periodic model aggregation. Therefore, incorporating FL into the design of the next generation of ML-oriented data marketplaces becomes highly desirable as it mitigates privacy concerns for data owners. Nevertheless, recent studies (e.g., [9, 19, 36, 38, 47, 53]) have uncovered privacy threats that data owners participating in FL are still vulnerable to. Specifically, honest but curious model requesters or external adversaries may infer data owners' private or sensitive information from their shared updated model parameters or intermediate gradients during the FL process.

To safeguard data owners against such privacy threats in FL, differential privacy (DP) techniques have been widely incorporated due to their rigorous privacy guarantee and easy implementation [24, 32, 40, 43, 46]. This integration has given rise to the paradigm of differentially private federated learning (DPFL). In each round of DPFL, each participating data owner will obfuscate their computed stochastic gradients or local model updates before sharing them according to their affordable privacy budgets.[1] In this way, it is more difficult for untrusted third parties to compromise the privacy of data owners, thereby ensuring enhanced privacy preservation. Nonetheless, it should be noted that data owners with moderate privacy budgets, which involve introducing mild noise perturbations, still sustain a certain degree of potential privacy disclosure. Consequently, they bear some privacy costs associated with potential negative consequences stemming from privacy loss [12, 32, 33]. For example, data owners may experience privacy loss due to exposed demographic information, browsing history, and purchasing patterns. This could make them vulnerable to price discrimination, ultimately resulting in financial losses.

To facilitate privacy-preserving data sharing for ML applications, researchers have developed several data marketplaces with DPFL (e.g., [12, 33, 45]). As illustrated in Figure 1, these data marketplaces consist of two main components: data owners and model requesters. Model requesters, such as facial recognition service providers, seek to engage data owners possessing relevant training data (e.g., facial images) to contribute to their model training tasks through DPFL. These interactions can be conceptualized as model requesters purchasing the training data from data owners in *a*

[1]The privacy budget refers to a data owner's maximum tolerable privacy loss. A higher privacy budget indicates that the data owner can tolerate more privacy loss.

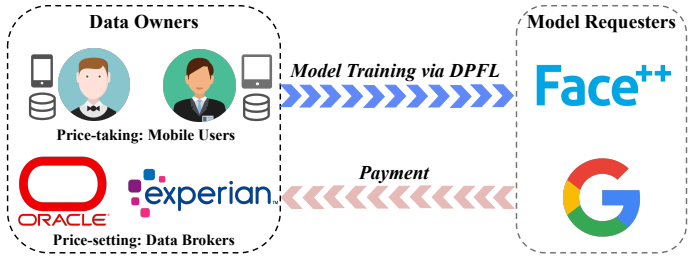

Fig. 1. A DPFL-based data marketplace, where model requesters recruit data owners to contribute to their model training tasks via DPFL.

usable but invisible manner. Hence, DPFL-based data marketplaces are more practical than escrow-based ones, where raw training data is exposed. In DPFL-based data marketplaces, data owners implement gradient (model) perturbation to mitigate potential privacy risks with a specific noise level determined by their allocated privacy budget. Simultaneously, model requesters are required to provide sufficient compensation to each participating data owner to offset their incurred privacy costs during the DPFL process.

1.2 Motivations and Contributions

Existing designs for DPFL-based data marketplaces (e.g., [12, 33, 45, 49]) generally assume a leader-follower model, where *the model requester, who can be AI service providers like Google, is presumed to possess greater market power than data owners, who are typically mobile users like Google Gboard users*. Inspired by the Stackelberg game, where the leader moves first and then the followers move sequentially, some studies (e.g., [12, 45]) formulated the market as a two-stage Stackelberg game, with the model requester taking the role of the leader, setting prices, while data owners act as price-taking followers. Other studies (e.g., [33, 49]) modeled the market as a procurement auction, where the model requester acts as the auctioneer determining payments for winning data owners. However, *practical scenarios usually involve another class of data owners, i.e., data brokers with a substantial amount of diversified data* [25, 39], like Experian[2], Equifax[3], Acxiom[4], and Epsilon[5]. These data owners possess significant market power and can set prices rather than merely accept them.[6] In such cases, the aforementioned existing models are no longer applicable for operating such markets. Therefore, it is imperative to explore the optimal approach for the model requester to interact with both price-taking and price-setting data owners in a DPFL-based data marketplace.

Building upon the above discussions, in this work, we advocate a novel DPFL-based data marketplace that enables the model requester to interact with both price-setting and price-taking data owners. We consider individual mobile users or small institutions as price-taking data owners. The model requester is prioritized to determine market prices for this type of data owners. On the other hand, price-setting data owners are those who have more market power than the model requester. For instance, large institutions like hospitals and giant companies that own a tremendous amount

[2]https://www.experian.com.sg/
[3]https://www.equifax.com/
[4]https://www.acxiom.com/
[5]https://www.epsilon.com/us
[6]In this work, we capture the market power of different types of data owners based on their capability to set prices or merely accept prices proposed by others. In our future work, we will incorporate more practical factors to comprehensively characterize market power.

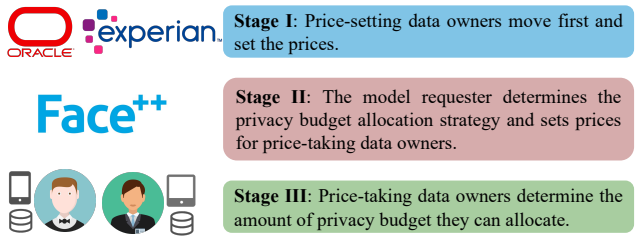

Fig. 2. A three-stage Stackelberg game framework for a DPFL-based data marketplace where a model requester interacts with both price-setting and price-taking data owners.

of valuable data may have more market power than a single model requester. Such a marketplace setting can accommodate various real-world applications, two of which are provided below.

- *Healthcare*: In this application, hospitals and medical institutions with large patient databases are price-setting data owners, while individual patients or smaller healthcare providers are price-taking data owners. The model requester, such as a pharmaceutical company or research organization, can use our marketplace to purchase data from these two types of data owners for training ML models that aid in drug development or medical treatments.
- *Finance*: In this application, major banks and financial institutions with transactional and customer data serve as price-setting data owners, while smaller banks, credit unions, or individual consumers act as price-taking data owners. The model requester, such as a regulatory agency or fintech company, can leverage our data marketplace to acquire data from these two types of data owners for training models in fraud detection, risk assessment, or customer behavior analysis, while ensuring privacy.

Our work stands apart from existing literature in the following two fundamental aspects:

- The model requester negotiates with both price-setting and price-taking data owners.
- The model requester's primary objective is to maximize profit, which is the difference between the utility obtained from DPFL-based model training and the payment disbursed to data owners.

To facilitate the analysis, we model the interactions between the model requester and the two types of data owners as a three-stage Stackelberg game, as depicted in Figure 2. Within this framework, we study the model requester's profit maximization problem. Through the formulated game, we investigate the price competition among price-setting data owners and the competition among price-taking data owners for the allocation of privacy budget to the DPFL-based model training task requested by the model requester.

- In the first stage, price-setting data owners strategically set their prices to maximize their individual payoffs.
- Moving to the second stage, the model requester decides whether to engage the price-setting data owners by either accepting or declining their proposed prices. In acceptance cases, the model requester also determines the quantity of privacy budget to purchase. Furthermore, during this stage, the model requester sets prices for the price-taking data owners to maximize its profit.
- Finally, in the third stage, price-taking data owners follow the model requester's prices to determine how much privacy budget they will consume, aiming to maximize their individual payoffs.

To summarize, we make the following contributions:

- ***A Novel DPFL-based Data Marketplace***: To the best of our knowledge, we propose the first data marketplace with differentially private federated learning that accommodates both price-setting

and price-taking data owners. We model the interactions between the model requester and the two types of data owners as a three-stage Stackelberg game. Besides, we derive the corresponding subgame perfect equilibrium, which provides insights into market prices.

- ***Rigorous Equilibrium Analysis***: We conduct a rigorous game analysis, demonstrating its convex nature and proving the existence of a subgame perfect equilibrium. Moreover, we demonstrate that the model requester's profit maximization problem leads to a unique equilibrium strategy.
- ***Efficient Algorithm Design***: To obtain the subgame perfect equilibrium, we propose iterative algorithms, one for price-setting data owners to determine their best response strategies and one for the model requester to facilitate information exchange among the data owners. Employing the proposed algorithms, the price-setting data owners do not need to obtain full information about the model requester and other data owners. We prove that the proposed algorithms converge to the unique subgame perfect equilibrium strategy.
- ***Extensive Performance Evaluation***: Extensive numerical results demonstrate that the proposed three-stage market model improves the model requester's profit compared to the case where the model requester only recruits price-taking data owners for model training. We also show that a higher model quality requirement increases data owners' payoffs. Furthermore, we validate the rapid convergence of our proposed iterative algorithms to the unique subgame perfect equilibrium strategy.

1.3 Literature Review

1.3.1 ML-oriented Data Marketplaces. An ML-oriented data marketplace generally involves interactions between model requesters (i.e., data buyers) and data owners (i.e., data sellers, such as individuals or organizations). Agarwal *et al.* in [1] proposed an algorithmic solution for a two-sided data marketplace to efficiently buy and sell training data for ML tasks. Chen *et al.* in [6] proposed a model-based pricing framework for pricing ML models in a data marketplace. Liu *et al.* in [20] introduced the first end-to-end model marketplace with differential privacy, which formulates compensation functions for data owners and pricing functions for model buyers. Xu *et al.* in [42] designed the first online data valuation and pricing mechanism for ML tasks in mobile health systems, which incentivizes users to contribute their m-health data by determining payments based on data valuation. Chen *et al.* in [5] designed a class of mechanisms that price data via costly signaling to address the challenge of pricing data for machine learners when neither the seller nor the machine learner knows the true quality of the data.

The aforementioned data marketplaces require data owners to exchange their raw training data with model requesters or brokers for centralized model training. In contrast, our design incorporates data owners in the model training process through FL, effectively alleviating their privacy concerns. Furthermore, we enable data owners to implement gradient perturbation during FL (yielding DPFL) to defend against advanced inference attacks.

1.3.2 DPFL-based Data Marketplace Design. Recent years have witnessed the development of data marketplaces based on DPFL. For instance, Zheng *et al.* in [52] proposed FL-Market, a locally private model marketplace that ensures trustworthy data acquisition for ML-based data analytics. Zhang *et al.* in [49] addressed both protection against malicious participants and fair compensation for clients contributing their data and resources to DPFL-based model training. Hu *et al.* in [12] formulated a *two-stage* Stackelberg game to model the interactions between clients and the server, aiming to maximize the server's profit. Yi *et al.* in [45] introduced a Stackelberg incentive mechanism for wireless FL with DP, effectively encouraging mobile devices to participate in DPFL. Sun *et al.* in [33] proposed a profit-maximizing model marketplace with DPFL, where the broker incentivizes data owners to train a model and subsequently sells the model to model consumers.

Our work differs from the above studies in two fundamental aspects. First, the model requester in our work negotiates with both price-setting and price-taking data owners. This aspect introduces a novel dynamic where the model requester interacts with data owners who have the ability to set their own prices. Second, the model requester's primary objective is to maximize profit, which is the difference between the utility obtained from DPFL-based model training and the payment disbursed to data owners. These two aspects substantially increase the challenge of formally characterizing the interactions between data owners and the model requester.

2 Preliminaries

2.1 Federated Learning

FL enables collaborative model training involving multiple data owners without collecting their raw training data. Suppose there is a set $\mathcal{W} = \{1, \ldots, W\}$ of data owners. Each data owner $w \in \mathcal{W}$ possesses a private training dataset $\mathcal{D}_w = \left\{\left(\boldsymbol{\tau}_1^w, y_1^w\right), \ldots, \left(\boldsymbol{\tau}_{D_w}^w, y_{D_w}^w\right)\right\}$ containing $|\mathcal{D}_w| = D_w$ data samples, where $\boldsymbol{\tau}_i^w$ and y_i^w represent the features and the corresponding label of data sample i associated with data owner w, respectively. Let $\boldsymbol{\theta} \in \mathbb{R}^d$ denote the model parameters and $\ell(\boldsymbol{\tau}, y; \boldsymbol{\theta})$ represent the sample-wise loss value. Then, for each data owner $w \in \mathcal{W}$, the local loss function is computed as $f_w(\boldsymbol{\theta}) = \frac{1}{D_w} \sum_{i=1}^{D_w} \ell\left(\boldsymbol{\tau}_i^w, y_i^w; \boldsymbol{\theta}\right)$. The global loss function is subsequently defined as the average of the local loss functions of all data owners: $f(\boldsymbol{\theta}) = \frac{1}{W} \sum_{w \in \mathcal{W}} f_w(\boldsymbol{\theta})$. FL aims to derive an optimal $\boldsymbol{\theta}$ through minimizing $f(\boldsymbol{\theta})$ [27].

2.2 Differential Privacy

A randomized algorithm $\mathcal{A}$ is considered to satisfy DP when, for two adjacent input datasets of $\mathcal{A}$, differing by at most one data record, the algorithm generates statistically indistinguishable outputs [8]. A widely adopted concept to quantify privacy loss in machine learning algorithms is (ϵ, δ)-differential privacy [18, 21]. In this work, we employ a relaxed version of (ϵ, δ)-differential privacy, i.e., ρ-zero-concentrated differential privacy (henceforth ρ-zCDP) developed in [3] for privacy analysis. ρ-zCDP offers a more stringent and precise composition bound [3], facilitating a more accurate and nuanced privacy analysis, particularly when dealing with iterative algorithms.

To formally define ρ-zCDP, we introduce a random variable Y. Consider a randomized algorithm $\mathcal{A}$, which maps from a domain $\mathcal{D}$ to a range $\mathcal{R}$. For any two adjacent input datasets $D, D' \in \mathcal{D}$ and any subset of outputs $\mathcal{S} \in \mathcal{R}$, the privacy loss Y of $\mathcal{A}$ is defined as $Y = \log \frac{\Pr[\mathcal{A}(D)=\mathcal{S}]}{\Pr[\mathcal{A}(D')=\mathcal{S}]}$. Then, ρ-zCDP is formally defined in Definition 1.

Definition 1. *(ρ-zCDP [3]): A randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{R}$ satisfies ρ-zCDP if for any $\gamma \in (1, \infty)$, we have*

$$\mathbb{E}\left[e^{(\gamma-1)Y}\right] \leq e^{(\gamma-1)(\rho\gamma)}, \tag{1}$$

where $\mathbb{E}$ represents the expectation operator applied to Y.

The above (1) implies that Y follows a *subGaussian* distribution with a mean of ρ and a variance of 2ρ [3]. Thus, smaller values of $\rho > 0$ correspond to lower privacy loss with a higher probability. *In this work, we employ the parameter ρ in ρ-zCDP to quantify a data owner's privacy budget.* In Lemma 1 below, we show a popular mechanism that can achieve ρ-zCDP. This result will be used for the convergence analysis of DPFL in Section 3.2.

Lemma 1 (Gaussian Mechanism [3]). *Let $h : \mathcal{D} \to \mathbb{R}$ be any real-valued function with sensitivity $\Delta_h = \max_{D,D'} |h(D) - h(D')|$ for any two adjacent datasets $D, D' \in \mathcal{D}$. The Gaussian mechanism $\mathcal{A}$, which returns $\mathcal{A}(D) = h(D) + \mathbb{N}(0, \sigma^2)$, satisfies $(\Delta_h^2/2\sigma^2)$-zCDP.*

2.3 Differentially Private Federated Learning

To thwart privacy threats in FL, one can incorporate DP, yielding the framework of DPFL. Each round of DPFL has two major steps: 1) Local model training with gradient perturbation at the data owner side and 2) Global model update at the model requester side.

(1) Local Model Training with Gradient Perturbation. In a specific round indexed by $r \in \{1, 2, \ldots, R\}$, data owner $w \in \mathcal{W}$ receives the latest global model $\boldsymbol{\theta}^{r-1}$ from the model requester, where $\boldsymbol{\theta}^0$ is a random initialization. In this round, data owner w undertakes two essential steps:

- *Gradient Computation:* Data owner w first computes a stochastic gradient, which is denoted as $g\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right) = 1/B_w \sum_{\left(\boldsymbol{\tau}_i^w, y_i^w\right) \in \mathcal{B}_w} \nabla \ell\left(\boldsymbol{\tau}_i^w, y_i^w; \boldsymbol{\theta}^{r-1}\right)$, with a randomly sampled mini-batch $\mathcal{B}_w$ of size B_w from his or her local training dataset $\mathcal{D}_w$ (i.e., $\mathcal{B}_w \subseteq \mathcal{D}_w$).
- *Gradient Perturbation:* Data owner w then employs the Gaussian mechanism to implement gradient perturbation. Specifically, he or she injects Gaussian noises $\boldsymbol{b}_w^r \sim \mathbb{N}\left(0, \sigma_w^2 \mathbf{I}_d\right)$ ($\mathbf{I}_d$ represents the d-dimensional identity matrix, where d denotes the number of parameters in $\boldsymbol{\theta}$) into $g\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right)$, leading to the perturbed stochastic gradient $\tilde{g}\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right)$. Formally, we have

$$\tilde{g}\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right) = g\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right) + \boldsymbol{b}_w^r. \tag{2}$$

Following this, we present Lemma 2 to illustrate how the privacy of data owners is guaranteed when they implement gradient perturbation according to (2) during their participation in the DPFL-based model training task for the model requester.

Lemma 2 (Privacy Guarantee [32]). *In each round of DPFL, if data owner w performs gradient obfuscation following* (2)*, they are guaranteed with ρ_w-zCDP, where $\rho_w = \frac{2L^2}{B_w^2 \sigma_w^2}$ and L is the Lipschitz constant of the L-Lipschitz smooth loss function $f_w(\boldsymbol{\theta})$.*

(2) Global Model Update. After receiving the perturbed stochastic gradients $\tilde{g}\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right)$ from each data owner $w \in \mathcal{W}$, the model requester averages them and updates the global model with a learning rate η as follows:

$$\boldsymbol{\theta}^r = \boldsymbol{\theta}^{r-1} - \frac{\eta}{W} \sum\nolimits_{w \in \mathcal{W}} \tilde{g}\left(\mathcal{B}_w; \boldsymbol{\theta}^{r-1}\right). \tag{3}$$

3 System model

In this section, we first provide a high-level system overview in Section 3.1. Then, we characterize the data owners' payoff and the model requester's profit in Section 3.2. Finally, we formally formulate the problem in Section 3.3.

3.1 System Overview

As shown in Figure 2, we consider a data marketplace that consists of a model requester and a set $\mathcal{W} = \{1, \ldots, W\}$ of relevant data owners (including a set $\mathcal{W}^s = \{1, \ldots, W^s\}$ of price-setting data owners and a set $\mathcal{W}^t = \{1, \ldots, W^t\}$ of price-taking data owners). Formally, we have $\mathcal{W} = \mathcal{W}^s \cup \mathcal{W}^t$ and $\mathcal{W}^s \cap \mathcal{W}^t = \emptyset$. Each data owner $w \in \mathcal{W}$ has a private training dataset $\mathcal{D}_w$ (containing D_w data samples), which can be used to train the model desired by the model requester. The involved entities are detailed as follows:

- ***Model Requester:*** Before implementing the DPFL-based data marketplace, the model requester needs to identify the set $\mathcal{W}$ of relevant data owners who possess valuable training data. Several methods can be employed for this purpose:
  - *Data Owner Profiling*: The model requester can require data owners to create profiles that include details about the types of data they possess and the size of their datasets.

- *Learning from Others*: The model requester can gather insights from other model requesters who have previously interacted with the data owners.
- *Data Brokers*: In certain cases, data brokers connect model requesters with relevant data owners. These intermediaries know the available data sources and can help identify data owners that can satisfy the requirements of the model requester.

We consider an honest but curious model requester who seeks to recruit each data owner $w \in \mathcal{W}$ to contribute to his or her model (denoted as $\boldsymbol{\theta} \in \mathbb{R}^d$) training task, but may intend to infer data owners' private information.[7] Therefore, data owners participate in model training via DPFL to mitigate privacy risks. To incentivize data owners' participation, the model requester must provide adequate economic incentives (i.e., monetary payments), which are designed to compensate data owners for their incurred privacy costs during DPFL.

- ***Data Owners:*** Data owners should satisfy several requirements to implement the DPFL-based data marketplace in practice. First, data owners should possess relevant and valuable data for model training. The model requester can assess data relevance by requesting data owners to perform several rounds of local training and then evaluate the submitted local models on a small public validation dataset to see if there is a large decrease in the loss value. This approach is based on previous works that detect poisoned local models submitted by malicious clients [10, 41]. Second, data owners do not collude and do not act as adversaries. In particular, there are two types of data owners according to their market power.
  - ***Price-taking Data Owners:*** Price-taking data owners are typically individuals (e.g., mobile users). They have limited market power, and thus, the model requester sets the prices and corresponding payments for them.
  - ***Price-setting Data Owners:*** Price-setting data owners are typically entities such as data brokers (e.g., Acxiom [12]). They have large market power and can set their own prices for privacy budget allocation to DPFL-based model training tasks initiated by the model requester.

3.2 Data Owner and Model Requester Modeling

3.2.1 ***Data Owners' Payoff.*** Each data owner's payoff is determined by the *privacy cost* (incurred when participating in DPFL) and the *payment* (received from the model requester).

- ***Privacy Cost***. Data owners ought to allocate a certain amount of privacy budget to DPFL-based model training, corresponding to moderate gradient perturbation, to train a high-usability ML model for the model requester. Accordingly, data owners still sustain a certain degree of potential privacy disclosure, incurring some privacy (leakage) costs.
  Let x_w represent the privacy budget of data owner w allocated to each round of DPFL. Here, x_w determines the noise variance of the Gaussian mechanism (recall Lemma 2) adopted for gradient perturbation in (2). Accordingly, the privacy cost of data owner w is an increasing function of x_w. Specifically, a higher privacy budget implies milder gradient perturbation (i.e., a smaller value of σ_w^2), resulting in higher privacy costs. Additionally, the privacy cost also depends on data owner w's individual privacy preference α_w. The privacy preference of a data owner represents their personal valuation of privacy and measures their sensitivity to potential privacy leakage. It is important to note that the privacy preference is not an externally set parameter but is determined by the data owners themselves. Essentially, privacy preferences are considered private information of data owners, necessitating a truthful mechanism to elicit this information [4, 32].

[7] In this work, we assume that all relevant data owners are simultaneously online during the DPFL process. This assumption is both feasible and reasonable in many scenarios. For example, in finance applications involving major banks as price-setting data owners and smaller banks as price-taking data owners, both types of data owners participate in model training through cross-silo FL. This process typically involves a limited number of data owners, leading to a situation where data owners are generally online simultaneously throughout the entire learning process [13, 22].

Formally, the privacy cost of data owner w is represented as $c_w\left(x_w, \alpha_w\right)$. Following prior studies [12, 37], this work employs a quadratic function (note that the results and insights developed in this work also apply to other forms of strictly convex privacy cost functions) to characterize $c_w\left(x_w, \alpha_w\right)$. Specifically, for price-taking and price-setting data owners, their respective privacy costs are computed as:

$$c_w^t\left(x_w^t\right) = \alpha_w^t\left(x_w^t\right)^2 \text{ and } c_w^s\left(x_w^s\right) = \alpha_w^s\left(x_w^s\right)^2. \tag{4}$$

- ***Payment***. Data owners should receive proper payments from the model requester to compensate for their privacy costs. A price-setting data owner can determine such a payment, whereas a price-taking data owner needs to decide whether to accept or decline the model requester's determined payment. While payment functions are typically assumed to be linear in terms of demand (privacy budget in this work), some studies have explored nonlinear payment functions based on demand (e.g., [50]). This paper employs the following distinct payment functions to reflect the varying market power of different data owners:
  - *Price-setting data owners:* The model requester's payment function for a price-setting data owner $w \in \mathcal{W}^s$ is $q_w^s\left(x_w^s\right) = p_w^s\left(x_w^s\right)^2$, where p_w^s represents the price set by the data owner.
  - *Price-taking data owners:* The model requester's payment function for a price-taking data owner $w \in \mathcal{W}^t$ is $q_w^t\left(x_w^t\right) = p_w^t x_w^t$, with p_w^t signifying the price set by the model requester.

  The quadratic payment function associated with price-setting data owners reflects their market power, emphasizing a desire for more compensation since they act as leaders for the model requester. Conversely, for price-taking data owners, the linear payment function expresses the model requester's intent to minimize payment since the model requester acts as the leader for these data owners and has more market power to set the prices. Note that the market prices are not externally set parameters but are determined through solving the profit/payoff maximization problems by different entities in the data market (which will be formally formulated in Section 4).
- ***Payoff***. The payoffs for price-taking and price-setting data owners are as follows:

$$V_w^t\left(x_w^t, p_w^t\right) = p_w^t x_w^t - \alpha_w^t\left(x_w^t\right)^2, w \in \mathcal{W}^t, \tag{5}$$

$$V_w^s\left(x_w^s, p_w^s\right) = p_w^s\left(x_w^s\right)^2 - \alpha_w^s\left(x_w^s\right)^2, w \in \mathcal{W}^s, \tag{6}$$

where $p_w^s \geq \alpha_w^s$.

3.2.2 ***Model Requester's Profit***. The model requester's profit is determined by the *utility* (obtained from the trained model and dependent on model quality) and the *payment* (made to data owners).

(1) Model Quality Estimation: To estimate model quality, we must ascertain how the privacy budgets $\boldsymbol{x} \triangleq (x_w : \forall w \in \mathcal{W})$ influence the convergence behavior of DPFL. We start with several assumptions commonly adopted in the literature [2, 12, 32]. Assumption 1 ensures that the gradient of $f_w(\boldsymbol{\theta})$ does not change arbitrarily quickly with respect to $\boldsymbol{\theta}$. Assumption 2 states that, in expectation, the vector $-g(\mathcal{B}_w; \boldsymbol{\theta}^r)$ is a direction of sufficient descent for $f(\boldsymbol{\theta})$ and the variance of $g(\mathcal{B}_w; \boldsymbol{\theta}^r)$ is bounded.

Assumption 1. *The loss function $f_w(\boldsymbol{\theta})$ at each data owner $w \in \mathcal{W}$ is L-Lipschitz smooth with Lipschitz constant $L > 0$.*

Assumption 2. *For all $r \in \mathbb{N}$, and for each data owner $w \in \mathcal{W}$, the global loss function $f(\boldsymbol{\theta}^r)$ is lower bounded by f^* and the stochastic gradients satisfy: (a) $\nabla f(\boldsymbol{\theta}^r)^T \mathbb{E}[g(\mathcal{B}_w; \boldsymbol{\theta}^r)] \geq \mu \|\nabla f(\boldsymbol{\theta}^r)\|_2^2$; (b)*

$\|\mathbb{E}[g(\mathcal{B}_w;\boldsymbol{\theta}^r)]\|_2 \leq \mu_G\|\nabla f(\boldsymbol{\theta}^r)\|_2$; *and (c)* $\mathbb{V}[g(\mathcal{B}_w;\boldsymbol{\theta}^r)] \leq M + M_V\|\nabla f(\boldsymbol{\theta}^r)\|_2^2$, *where* $\mathbb{V}[\cdot]$ *computes the variance.*[8] *Constants* $\mu_G \geq \mu > 0$, $M \geq 0$, *and* $M_V \geq 0$ *are defined.*

Theorem 1 illustrates the convergence behavior of DPFL for a non-convex loss function $f(\boldsymbol{\theta})$. For the sake of brevity, the convergence analysis for strictly convex loss functions is omitted due to space constraints. It is important to note that the results and insights presented in this work also apply therein.

Theorem 1. *Under Assumptions 1 and 2, consider that the model requester's model* $\boldsymbol{\theta}$ *is updated following* (3) *with a fixed learning rate* η *satisfying* $0 < \eta \leq \frac{\mu}{L(\mu_G^2+M_V)}$. *Then, the expected average-squared* ℓ_2*-norm of gradients of* $f(\boldsymbol{\theta})$ *over* R *rounds of updates satisfies*

$$\mathbb{E}[\frac{1}{R}\sum_{r=1}^{R}\|\nabla f(\boldsymbol{\theta}^{r-1})\|_2^2] \leq \frac{2(f(\boldsymbol{\theta}^0)-f^*)}{R\eta\mu} + \Phi, \tag{7}$$

where $\Phi = L\eta\left(W^2M + 2dL^2\sum_{w\in\mathcal{W}} 1/\left(B_w^2 x_w\right)\right)/\left(\mu W^2\right)$.

Proof. Given Assumption 1 and with reference to [2], we have

$$f(\boldsymbol{\theta}_1) \leq f(\boldsymbol{\theta}_2) + \nabla f(\boldsymbol{\theta}_2)^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_2) + \frac{1}{2}L\|\boldsymbol{\theta}_1-\boldsymbol{\theta}_2\|_2^2,\ \forall\boldsymbol{\theta}_1,\boldsymbol{\theta}_2 \in \mathbb{R}^d. \tag{8}$$

Then, the iterates generated based on the update rule in (3) satisfy the following inequalities for all $r \in \mathbb{N}$:

$$f(\boldsymbol{\theta}^r) - f\left(\boldsymbol{\theta}^{r-1}\right) \leq \nabla f\left(\boldsymbol{\theta}^{r-1}\right)^T\left(-\frac{\eta}{W}\sum\nolimits_{w\in\mathcal{W}}\tilde{g}\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)\right) + \frac{1}{2}L\| - \frac{\eta}{W}\sum\nolimits_{w\in\mathcal{W}}\tilde{g}\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)\|_2^2. \tag{9}$$

Since the injected Gaussian noises $\boldsymbol{b}_w^r \sim \mathbb{N}\left(0,\sigma_w^2\mathbf{I}_d\right)$ at each data owner $w \in \mathcal{W}$ are independent, and $\boldsymbol{b}_w^r$ is independent of $\sum_{w\in\mathcal{W}} g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)$, we have

$$\mathbb{E}[\boldsymbol{b}_w^r] = \mathbf{0}, \tag{10}$$

$$\mathbb{E}[\|\sum\nolimits_{w\in\mathcal{W}}\boldsymbol{b}_w^r\|_2^2] = \sum\nolimits_{w\in\mathcal{W}}\mathbb{E}[\|\boldsymbol{b}_w^r\|_2^2] = d\sum\nolimits_{w\in\mathcal{W}}\sigma_w^2, \tag{11}$$

$$\mathbb{E}[(\boldsymbol{b}_w^r)^T\sum_{w\in\mathcal{W}} g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)] = \mathbb{E}[\boldsymbol{b}_w^r]^T\mathbb{E}[\sum_{w\in\mathcal{W}} g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)] = 0. \tag{12}$$

Taking expectations in (9) *w.r.t* the distribution of $\mathcal{B}_w$ yields

$$\begin{aligned}\mathbb{E}[f(\boldsymbol{\theta}^r)] - f\left(\boldsymbol{\theta}^{r-1}\right) \leq & -\frac{\eta}{W}\nabla f\left(\boldsymbol{\theta}^{r-1}\right)^T\left(\sum\nolimits_{w\in\mathcal{W}}\mathbb{E}[g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)] + \sum\nolimits_{w\in\mathcal{W}}\mathbb{E}[\boldsymbol{b}_w^r]\right)\\ & + \frac{L\eta^2}{2W^2}\left(\mathbb{E}[\|\sum\nolimits_{w\in\mathcal{W}} g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)\|_2^2] + \mathbb{E}[\|\sum\nolimits_{w\in\mathcal{W}}\boldsymbol{b}_w^r\|_2^2]\right)\\ & + \frac{L\eta^2}{W^2}\mathbb{E}[\sum\nolimits_{w\in\mathcal{W}}(\boldsymbol{b}_w^r)^T\sum\nolimits_{w\in\mathcal{W}} g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)].\end{aligned} \tag{13}$$

[8]We would like to clarify that the bound on variance here is only associated with *gradient error*, which is different from *gradient noise*. Specifically, the stochastic gradient $g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)$ computed at each data owner w in round r is based on the current global model $\boldsymbol{\theta}^{r-1}$ and *a randomly sampled mini-batch* $\mathcal{B}_w$ from the training data. Thus, the computed stochastic gradient $g\left(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}\right)$ deviates from the actual gradient $\nabla f\left(\boldsymbol{\theta}^{r-1}\right)$ at a certain distance, which is referred to as *gradient error*. In contrast, the privacy budget determines the variance of the Gaussian noise added to the already computed stochastic gradient, regardless of the *gradient error* (recall Lemma 2).

Then, based on (10), (11), and (12), we have

$$\begin{aligned}\mathbb{E}[f(\boldsymbol{\theta}^r)] - f(\boldsymbol{\theta}^{r-1}) \leq & -\frac{\eta}{W}\nabla f(\boldsymbol{\theta}^{r-1})^T \left(\sum\nolimits_{w\in\mathcal{W}} \mathbb{E}[g(\mathcal{B}_w;\boldsymbol{\theta}^{r-1})]\right) \\ & + \frac{L\eta^2}{2W^2}\left(\mathbb{E}[\| \sum\nolimits_{w\in\mathcal{W}} g(\mathcal{B}_w;\boldsymbol{\theta}^{r-1}) \|_2^2] + d\sum\nolimits_{w\in\mathcal{W}} \sigma_w^2\right).\end{aligned} \tag{14}$$

Based on Assumption 2 and the fundamental inequalities, and defining $M_G = \mu_G^2 + M_V$, we have

$$\begin{aligned}\mathbb{E}[f(\boldsymbol{\theta}^r)] - f(\boldsymbol{\theta}^{r-1}) &\leq \frac{L\eta^2}{2W^2}\left(W^2M + d\sum\nolimits_{w\in\mathcal{W}}\sigma_w^2\right) - (\eta\mu - \frac{L\eta^2 M_G}{2})\|\nabla f(\boldsymbol{\theta}^{r-1})\|_2^2 \\ &\leq -\frac{1}{2}\eta\mu\|\nabla f(\boldsymbol{\theta}^{r-1})\|_2^2 + \frac{L\eta^2}{2W^2}\left(W^2M + d\sum\nolimits_{w\in\mathcal{W}}\sigma_w^2\right).\end{aligned} \tag{15}$$

Summing both sides of the above inequality for $r \in \{1,\ldots,R\}$ and recalling Assumption 2(a) gives

$$f^* - f(\boldsymbol{\theta}^0) \leq \mathbb{E}[f(\boldsymbol{\theta}^R)] - f(\boldsymbol{\theta}^0) \leq -\frac{1}{2}\eta\mu\mathbb{E}\left[\sum_{r=1}^{R}\|\nabla f(\boldsymbol{\theta}^{r-1})\|_2^2\right] + \frac{RL\eta^2}{2W^2}\left(W^2M + d\sum\nolimits_{w\in\mathcal{W}}\sigma_w^2\right). \tag{16}$$

Rearranging yields

$$\mathbb{E}\left[\frac{1}{R}\sum_{r=1}^{R}\|\nabla f(\boldsymbol{\theta}^{r-1})\|_2^2\right] \leq \frac{2(f(\boldsymbol{\theta}^0) - f^*)}{R\eta\mu} + \frac{L\eta(W^2M + d\sum_{w\in\mathcal{W}}\sigma_w^2)}{\mu W^2}, \tag{17}$$

which is equivalent to (7) in Theorem 1 given Lemma 2.

Here the proof is complete. □

For non-convex model training via DPFL, quantifying the actual loss value of the model at convergence is challenging due to the inherent non-convexity of the optimization problem. However, the ℓ_2-norm of gradients plays a crucial role in assessing the optimization progress in non-convex optimization. It serves as a widely accepted metric for gauging the steepness or slope of the loss function at a specific point within the parameter space. Reaching a local minimum (instead of the global minimum) is often a practical objective for non-convex optimization. Lower ℓ_2-norm values of gradients indicate the model parameters are in close proximity to a local minimum of the loss function [15]. In practical scenarios, this proximity is generally associated with better model quality.

Therefore, our established upper bound for the expected average-squared ℓ_2-norm of gradients of $f(\boldsymbol{\theta})$ in Theorem 1, denoted as $2\left(f\left(\boldsymbol{\theta}^0\right) - f^*\right)/(R\eta\mu) + \Phi$, can serve as a practical metric for assessing model quality in DPFL. To enhance training convergence and attain higher model quality, it is imperative to minimize this upper bound. Its first component, $2\left(f\left(\boldsymbol{\theta}^0\right) - f^*\right)/(R\eta\mu)$, diminishes as the number of training rounds R increases, approaching zero. Hence, the expected average-squared ℓ_2-norm of gradients of $f(\boldsymbol{\theta})$ converges to a non-zero component denoted as Φ, which is linked to the privacy budgets allocated by data owners. Specifically, with a fixed data owner set $\mathcal{W}$, the model quality depends on $\psi = \sum_{w\in\mathcal{W}}(1/x_w)$ (other parameters within Φ are constant and unrelated to data owners).[9] Given our analysis that a lower norm of gradients indicates proximity to a (local) minimum, smaller values of ψ indicate superior model quality.

(2) Utility Characterization: Based on the analysis above, the model quality is a function of $\boldsymbol{x}$. We use $u(\boldsymbol{x}) = \psi_0 - \psi = \psi_0 - \sum_{w\in\mathcal{W}}(1/x_w)$ to represent the utility of the model requester. Here $u(\boldsymbol{x})$ quantifies the decrease in the value of ψ, which reflects the improvement in model quality, in comparison to a significantly large constant ψ_0 that is associated with the minimum required

[9]We set the mini-batch size B_w as a constant across training rounds and data owners.

privacy budget by the model requester from each data owner. Since ψ is a strictly convex and decreasing function of x_w, $u(\boldsymbol{x})$ is a positive, increasing, and strictly concave function of $\boldsymbol{x}$.

(3) Profit Calculation: The model requester's profit is given as:

$$V(\boldsymbol{x}, \boldsymbol{p}) = \psi_0 - \sum_{w \in \mathcal{W}} 1/x_w - \sum_{w \in \mathcal{W}^s} p_w^s \left(x_w^s\right)^2 - \sum_{w \in \mathcal{W}^t} p_w^t x_w^t, \tag{18}$$

where $\boldsymbol{p} \triangleq \left(\boldsymbol{p}^t, \boldsymbol{p}^s\right)$.

3.3 DPFL under Price Competition Game

As illustrated in Figure 2, we model the interactions between the model requester and the two types of data owners as a three-stage Stackelberg game. All the players determine their strategies in each stage to maximize their payoff or profit.

- Stage I: *Player*: price-setting data owners $w \in \mathcal{W}^s$; *Strategy*: price vector $\boldsymbol{p}^s \triangleq \left(p_w^s : \forall w \in \mathcal{W}^s\right)$; *Payoff*: $V_w^s\left(x_w^s, p_w^s\right)$ given in (6).
- Stage II: *Player*: model requester; *Strategy*: privacy budget vector for price-setting data owners $\boldsymbol{x}^s \triangleq \left(x_w^s : \forall w \in \mathcal{W}^s\right)$ and price vector $\boldsymbol{p}^t \triangleq \left(p_w^t : \forall w \in \mathcal{W}^t\right)$ for price-taking data owners; *Profit*: $V(\boldsymbol{x}, \boldsymbol{p})$ given in (18).
- Stage III: *Player*: price-taking data owners $w \in \mathcal{W}^t$; *Strategy*: privacy budget vector $\boldsymbol{x}^t \triangleq \left(x_w^t : \forall w \in \mathcal{W}^t\right)$; *Payoff*: $V_w^t\left(x_w^t, p_w^t\right)$ given in (5).

We aim to determine this three-stage game's subgame perfect equilibrium (SPE), ensuring that neither the model requester nor the data owners have incentives to deviate unilaterally.

Definition 2 (Subgame Perfect Equilibrium). *A strategy profile $\left(\boldsymbol{x}^{NE}, \boldsymbol{p}^{NE}\right)$, including privacy budget strategies $\boldsymbol{x}^{NE}$ and pricing strategies $\boldsymbol{p}^{NE}$, is a subgame perfect equilibrium if it represents a Nash equilibrium (NE) in every subgame of the original game.*

In the next section, we will formally formulate the three-stage Stackelberg game. We will use backward induction [26] to obtain the corresponding SPE.

4 Three-Stage Game Formulation

4.1 Stage III (Price-taking Data Owners)

In Stage III, given the price-vector $\boldsymbol{p}^t$ set by the model requester, the price-taking data owners determine how much privacy budget they would allocate to DPFL-based model training to maximize their own payoffs. In particular, each data owner $w \in \mathcal{W}^t$ selects its strategy x_w^t within the strategy space $\left[\eta_{x_w^t}, \epsilon_{x_w^t}\right]$ (where $\eta_{x_w^t}$ is a minimum required privacy budget by the model requester to ensure a certain model quality level and $\epsilon_{x_w^t}$ is the maximum affordable privacy budget of data owner w) to maximize its payoff $V_w^t\left(x_w^t, p_w^t\right)$.[10] This leads to the following optimization problem:

$$\max_{x_w^t} \quad V_w^t\left(x_w^t, p_w^t\right) \tag{19a}$$

$$\text{s.t.} \quad \eta_{x_w^t} \le x_w^t \le \epsilon_{x_w^t}. \tag{19b}$$

As shown in (5), V_w^t is a quadratic function with a negative quadratic coefficient. Thus, V_w^t is a concave function. Besides, the feasible set given in (19b) is convex. Therefore, Problem (19) is a

[10]Without loss of generality, we consider a homogeneous minimum required privacy budget among data owners. This specific value can be determined from historical model training outcomes.

concave maximization problem. Let $x_w^{t^*}\left(p_w^t\right)$ denote the unique optimal solution of Problem (19), which is given in Theorem 2.

Theorem 2. *The optimal privacy budget of price-taking data owner w is*

$$x_w^{t^*}\left(p_w^t\right) = \begin{cases} \eta_{x_w^t}, & if \quad p_w^t < p_w^{t,\min} \\ \frac{p_w^t}{2\alpha_w^t}, & if \quad p_w^{t,\min} \le p_w^t \le p_w^{t,\max} \\ \epsilon_{x_w^t}, & if \quad p_w^t > p_w^{t,\max} \end{cases}, \tag{20}$$

where $p_w^{t,\min} = 2\alpha_w^t \eta_{x_w^t}$ and $p_w^{t,\max} = 2\alpha_w^t \epsilon_{x_w^t}$.

Proof. The Lagrangian of Problem (19) is presented as

$$L^t = p_w^t x_w^t - \alpha_w^t \left(x_w^t\right)^2 - \lambda_w^* \left(x_w^t - \epsilon_{x_w^t}\right) + \psi_w^* \left(x_w^t - \eta_{x_w^t}\right), \tag{21}$$

where λ_w^* and ψ_w^* are the optimal Lagrangian multipliers. Based on the Karush-Kuhn-Tucker (KKT) conditions, we consider the following different cases:

- $\lambda_w^* = 0$ and $\psi_w^* = 0$: In this case, the optimal solution satisfies $p_w^t - 2\alpha_w^t x_w^t = 0$, and we have $x_w^{t^*}\left(p_w^t\right) = p_w^t/\left(2\alpha_w^t\right)$.
- $\lambda_w^* > 0$: We get $x_w^{t^*} = \epsilon_{x_w^t}$. It means that the total privacy budget of data owner w is allocated to DPFL. This condition is satisfied when $p_w^t > p_w^{t,\max} = 2\alpha_w^t \epsilon_{x_w^t}$.
- $\psi_w^* > 0$: We get $x_w^{t^*} = \eta_{x_w^t}$. It means that the minimum privacy budget of data owner w is allocated to the DPFL-based model training. This condition is satisfied when $p_w^t < p_w^{t,\min} = 2\alpha_w^t \eta_{x_w^t}$.

□

Theorem 2 establishes that the optimal strategy $x_w^{t^*}\left(p_w^t\right)$ for the price-taking data owner w is a linear and increasing function in p_w^t within the interval $\left[p_w^{t,\min}, p_w^{t,\max}\right]$. Notice that the model requester will not set any price higher than $p_w^{t,\max}$ since the data owner cannot allocate a larger privacy budget than $\epsilon_{x_w^t}$. Furthermore, the model requester has a minimum required privacy budget $\eta_{x_w^t}$ from each data owner, which prevents setting an extremely low price. Specifically, the price cannot be lower than $p_w^{t,\min}$. Hence, the strategy space for the model requester is confined to $\left[p_w^{t,\min}, p_w^{t,\max}\right]$.

4.2 Stage II (Model Requester)

We now analyze the model requester's behavior in Stage II. The model requester's strategy space is $\mathcal{E} = \mathcal{E}_{\boldsymbol{x}^s} \times \mathcal{E}_{\boldsymbol{p}^t}$, where $\mathcal{E}_{\boldsymbol{x}^s} = \left\{\left[\eta_{x_w^s}, \epsilon_{x_w^s}\right]\right\}_{w \in \mathcal{W}^s}$ and $\mathcal{E}_{\boldsymbol{p}^t} = \left\{\left[p_w^{t,\min}, p_w^{t,\max}\right]\right\}_{w \in \mathcal{W}^t}$. Given the strategies $\boldsymbol{p}^s \triangleq \left(p_w^s : \forall w \in \mathcal{W}^s\right)$ of price-setting data owners and with the knowledge of how the model requester's strategy would affect price-taking data owners' strategies $x_w^{t^*}\left(p_w^t\right), w \in \mathcal{W}^t$, the model requester maximizes its own profit in Stage II. This leads to the following optimization problem:

$$\max_{(\boldsymbol{x}^s, \boldsymbol{p}^t) \in \mathcal{E}} \quad V\left(\left(\boldsymbol{x}^{t^*}\left(\boldsymbol{p}^t\right), \boldsymbol{x}^s\right), \left(\boldsymbol{p}^t, \boldsymbol{p}^s\right)\right) \tag{22a}$$

$$\text{s.t.} \quad \sum_{w \in \mathcal{W}^t} \frac{1}{x_w^{t^*}(p_w^t)} + \sum_{w \in \mathcal{W}^s} \frac{1}{x_w^s} \le S. \tag{22b}$$

- Objective function (22a) is to maximize model requester's profit.

- Constraint (22b) enforces a requirement on the model quality (recall Theorem 1 and relevant discussions).

The optimal solution to Problem (22) exists and is unique if the problem is a strictly concave maximization problem. Since (22b) is convex, the convexity of Problem (22) depends on the concavity of the objective function (22a). The following theorem proves that (22a) is concave.

Theorem 3. *Problem* (22) *is a strictly concave maximization problem under any fixed* $\boldsymbol{p}^s$ *and has a unique optimal solution.*

Proof. To investigate the concavity of (22a), we form the Hessian matrix $\boldsymbol{H}$ with the two sub-matrices $\boldsymbol{H}^{\boldsymbol{x}^s}$ and $\boldsymbol{H}^{\boldsymbol{x}^t}$ being the diagonal elements. Here, $\boldsymbol{H}^{\boldsymbol{x}^s} = \text{diag}\left(\partial^2 V/\partial(x_1^s)^2, \ldots, \partial^2 V/\partial(x_{W^s}^s)^2\right)$ and $\boldsymbol{H}^{\boldsymbol{x}^t} = \text{diag}\left(\partial^2 V/\partial(x_1^t)^2, \ldots, \partial^2 V/\partial(x_{W^t}^t)^2\right)$. Since $\partial^2 V/\partial(x_w^s)^2 = -2/(x_w^s)^3 - 2p_w^s \leq 0$ and $\partial^2 V/\partial(x_w^t)^2 = -2/(x_w^t)^3 - 4\alpha_w^t \leq 0$, $\boldsymbol{H}$ is negative semi-definite. Therefore, (22a) is concave, which completes the proof. □

The optimal solution $\left(\boldsymbol{x}^{s^*}\left(\boldsymbol{p}^s\right), \boldsymbol{x}^{t^*}\left(\boldsymbol{p}^s\right)\right)$ to Problem (22) depends on the strategy of price-setting data owners, i.e., $\boldsymbol{p}^s$, which will be obtained in Stage I.

4.3 Stage I (Price-setting Data Owners)

We now analyze the price competition among price-setting data owners, which determines their strategies (i.e., price vector $\boldsymbol{p}^s$) with the knowledge of how their strategies would affect the model requester's strategy. For price-setting data owner $w \in \mathcal{W}^s$, the optimal response of the model requester obtained in Stage II (i.e., $\boldsymbol{x}^{s^*}\left(\boldsymbol{p}^s\right)$) depends not only on price p_w^s, but also on the prices submitted by other players. This reflects the interdependence of data owners' pricing decisions. Let $\boldsymbol{p}_{-w}^s$ denote the strategy of the price-setting data owners excluding w. Thus, we have $\boldsymbol{p}^s = \left(p_w^s, \boldsymbol{p}_{-w}^s\right)$. Note that non-cooperative games are used in situations involving competition among the players, which is applicable for analyzing the price competition among price-setting data owners. Therefore, we form the price-setting non-cooperative game (PS-NCG) $\mathcal{G}^s\left(\mathcal{W}^s, \mathcal{E}^s, V_w^s\right)$, where $\mathcal{E}^s$ represents the strategy space. By substituting the optimal strategy $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ of the model requester obtained in Stage II into (6), the payoff function of price-setting data owner w is reformulated as

$$V_w^s\left(x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right), p_w^s\right) = \left(p_w^s - \alpha_w^s\right)\left(x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)\right)^2, \tag{23}$$

We first introduce the concept of best response strategy and NE to obtain the optimal strategies for price-setting data owners.

Definition 3 (Best Response Strategy [31]). *Given* $\boldsymbol{p}_{-w}^s$*, the best response strategy of data owner* w *is:*

$$p_w^{s^*} = \arg\max_{p_w^s \geq 0} \quad V_w^s\left(x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right), p_w^s\right) \tag{24a}$$

$$\text{s.t.} \quad \eta_{x_w^s} \leq x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) \leq \epsilon_{x_w^s}. \tag{24b}$$

Definition 4 (Nash Equilibrium [31]). *A strategy profile* $\boldsymbol{p}^{s^{\text{NE}}} = \left(p_1^{s^{\text{NE}}}, \ldots, p_{W^s}^{s^{\text{NE}}}\right)$ *is an NE if it is a fixed point of best responses, i.e., for all* $p_w^{s'} \geq 0, w \in \mathcal{W}^s$

$$V_w^s\left(x_w^{s^*}\left(p_w^{s^{\text{NE}}}, \boldsymbol{p}_{-w}^{s^{\text{NE}}}\right), p_w^{s^{\text{NE}}}\right) \geq V_w^s\left(x_w^{s^*}\left(p_w^{s'}, \boldsymbol{p}_{-w}^{s^{\text{NE}}}\right), p_w^{s'}\right).$$

To attain the best response strategies for price-setting data owners, we first need to know the model requester's optimal strategy $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ in response to the price vector $\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$. We determine the model requester's optimal strategy and its properties through Lemma 3 to analyze the existence and uniqueness of the NE in the following.

Lemma 3. *Given $\boldsymbol{p}_{-w}^s$, the model requester's optimal strategy is*

$$x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = \begin{cases} \epsilon_{x_w^s}, & p_w^s < p_w^{s,\min} \\ \eta_{x_w^s}, & p_w^s > p_w^{s,\max} \\ L_w^s\left(p_w^s, \boldsymbol{p}_{-w}^s\right), & p_w^{s,\min} \le p_w^s \le p_w^{s,\max} \end{cases}, \tag{25}$$

where $L_w^s\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ is a function of p_w^s and $\boldsymbol{p}_{-w}^s$. Besides, $p_w^{s,\min}$ and $p_w^{s,\max}$ are user dependent constants.

Proof. To derive the optimal strategy $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$, we construct the Lagrangian of Problem (22) as follows:

$$\begin{aligned} L = \psi_0 &- \sum\nolimits_{w\in\mathcal{W}} 1/x_w - \sum\nolimits_{w\in\mathcal{W}^s} p_w^s \left(x_w^s\right)^2 - \sum\nolimits_{w\in\mathcal{W}^t} p_w^t x_w^t \\ &- \sum\nolimits_{w\in\mathcal{W}^s} \lambda_w \left(x_w^s - \epsilon_{x_w^s}\right) + \sum\nolimits_{w\in\mathcal{W}^s} \psi_w \left(x_w^s - \eta_{x_w^s}\right) \\ &- \beta\left(\sum\nolimits_{w\in\mathcal{W}^t} 1/x_w^{t^*}(p_w^t) + \sum\nolimits_{w\in\mathcal{W}^s} 1/x_w^s - S\right), \end{aligned} \tag{26}$$

where λ_w, ψ_w, and β are Lagrangian multipliers.

We categorize data owners into $\mathcal{Q}_w^1 = \{m \in \mathcal{W} | x_m^* = \epsilon_{x_m^*}\}$, $\mathcal{Q}_w^2 = \{m \in \mathcal{W} | x_m^* = \eta_{x_m^*}\}$, and $\mathcal{Q}_w^3 = \{m \in \mathcal{W} | \eta_{x_m^*} < x_m^* < \epsilon_{x_m^*}\}$. Based on KKT conditions, we consider the following cases:

- If $\lambda_w > 0$, we have $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = \epsilon_{x_w^s}$ when $p_w^s \in \left[0, p_w^{s,\min}\right)$.
- If $\psi_w > 0$, we have $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = \eta_{x_w^s}$ when $p_w^s > p_w^{s,\max}$.
- If $\lambda_w = 0$, $\psi_w = 0$, and $\beta > 0$, we derive $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = L_w^s\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ with

$$L_w^s\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = \left(\sum_{m\in\mathcal{W}^t\cup\mathcal{Q}_w^3}\left(\frac{2\alpha_m^t}{p_w^s}\right)^{\frac{1}{3}} + \sum_{m\in\mathcal{W}^s\cup\mathcal{Q}_w^3}\left(\frac{p_m^s}{p_w^s}\right)^{\frac{1}{3}}\right)/S + x_w^*\left(\sum\nolimits_{m\in\mathcal{Q}_w^1}\frac{1}{\epsilon_{x_m^*}} + \sum\nolimits_{m\in\mathcal{Q}_w^2}\frac{1}{\eta_{x_m^*}}\right)/S. \tag{27}$$

- If $\lambda_w = 0$, $\psi_w = 0$, and $\beta = 0$, by substituting the solutions into (22a), the model requester's profit is smaller than the solutions derived when $\beta > 0$.

□

We now derive the following properties of the model requester's optimal strategy given in (25).

Lemma 4. *Given $\boldsymbol{p}_{-w}^s$, the model requester's optimal strategy $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ obtained in* (25) *is decreasing and convex in p_w^s for $p_w^s \in \left[p_w^{s,\min}, p_w^{s,\max}\right]$.*

Proof. By deriving the first and second derivatives from both sides of (25), we have $dx_w^{s^*}/dp_w^s \le 0$ and $d^2x_w^{s^*}/d\left(p_w^s\right)^2 \ge 0$. Therefore, we can conclude that $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ is decreasing and convex in p_w^s when $p_w^s \in \left[p_w^{s,\min}, p_w^{s,\max}\right]$. □

Remark 1. *Figure 3 illustrates the optimal solution $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ of the model requester, which validates Lemma 4. According to Figure 3, price-setting data owner w will never set its price $p_w^s < p_w^{s,\min}$, since the model requester procures the same amount of privacy budget (i.e., the maximum affordable privacy budget $\epsilon_{x_w^s}$ of data owner w) even when setting $p_w^s < p_w^{s,\min}$. In other words, the payoff function $V_w^s\left(x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right), p_w^s\right) < V_w^s\left(x_w^{s^*}\left(p_w^{s,\min}, \boldsymbol{p}_{-w}^s\right), p_w^{s,\min}\right)$ for any $p_w^s < p_w^{s,\min}$ since*

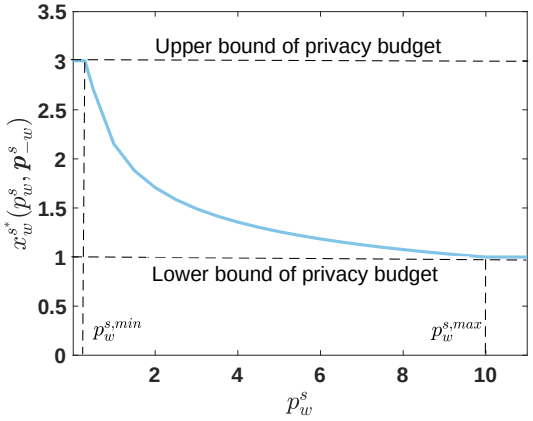

Fig. 3. The model requester's optimal strategy, which determines the amount of privacy budget $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ to be procured from price-setting data owner $w \in \mathcal{W}^s$.

$x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right) = x_w^{s^*}\left(p_w^{s,\min}, \boldsymbol{p}_{-w}^s\right)$. *Besides, price-setting data owner w should not set a price exceeding $p_w^{s,\max}$, as doing so would result in rejection by the model requester due to the minimum privacy budget requirement from each data owner. Excessively high prices lead to unacceptably high payments for the model requester. Therefore, we can limit the strategy space of price-setting data owner w to $p_w^s \in \left[p_w^{s,\min}, p_w^{s,\max}\right]$, where the model requester's optimal strategy $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ is decreasing and convex.*

5 Equilibrium Analysis

5.1 Existence of Nash Equilibria

We study PS-NCG formulated in Stage I to show whether an NE strategy exists. To derive the NE of the game, we can compute the price-setting data owners' best response strategies in this game by solving Problem (24). To this end, we first introduce a variable $\tilde{x}_w^s$ for each price-setting data owner $w \in \mathcal{W}^s$, and replace $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ by $\tilde{x}_w^s$. Furthermore, we replace p_w^s by $x_w^{s^*(-1)}\left(\tilde{\boldsymbol{x}}^s\right)$ in Problem (24), where $\tilde{\boldsymbol{x}}^s = \left(\tilde{x}_w^s : \forall w \in \mathcal{W}^s\right)$. The inverse function $x_w^{s^*(-1)}\left(\tilde{\boldsymbol{x}}^s\right)$ always exists due to Lemma 4. Therefore, the strategy space of data owner w is mapped one-to-one from $p_w^s \in \left[p_w^{s,\min}, p_w^{s,\max}\right]$ to $\tilde{x}_w^s \in \left[\eta_{x_w^s}, \epsilon_{x_w^s}\right]$. Hence, Problem (24) can be equivalently transformed into the following problem, from which the best response strategy of price-setting data owner w can be obtained.

$$p_w^{s^*} = \arg\max_{p_w^s \geq 0} \quad V_w^s\left(\tilde{x}_w^s, x_w^{s^*(-1)}\left(\tilde{\boldsymbol{x}}^s\right)\right) \tag{28a}$$

$$\text{s.t.} \quad \eta_{x_w^s} \leq \tilde{x}_w^s \leq \epsilon_{x_w^s}. \tag{28b}$$

For Problem (28), although the constraint (28b) is affine, the objective function may not be concave. We next prove the existence of an NE using Brouwer's fixed point theorem [31].

Theorem 4. *There exists an NE in Game PS-NCG.*

Proof. We first show that the fixed point solution of the best response strategies for all price-setting data owners exists. To determine the fixed point solution, we represent the data owner w's best response strategy obtained from Problem (28) as $\tilde{x}_w^s = f_w\left(\tilde{\boldsymbol{x}}_{-w}^s\right)$, where $\tilde{\boldsymbol{x}}_{-w}^s$ is the privacy

budget for the price-setting data owners excluding w. We further define the function $\tilde{\boldsymbol{x}}^s = \boldsymbol{F}(\tilde{\boldsymbol{x}}^s)$, where $\boldsymbol{F} = (f_w)_{w \in \mathcal{W}^s} : \left[\eta_{x_w^s}, \epsilon_{x_w^s}\right]_{w \in \mathcal{W}^s} \rightarrow \left[\eta_{x_w^s}, \epsilon_{x_w^s}\right]_{w \in \mathcal{W}^s}$. Brouwer's fixed point theorem [31] states that for any continuous function $\boldsymbol{F}$ that maps a closed convex set into itself, there is a point $\boldsymbol{x}_0$ such that $\boldsymbol{F}(\boldsymbol{x}_0) = \boldsymbol{x}_0$. Since the set $\left[\eta_{x_w^s}, \epsilon_{x_w^s}\right]$ is closed and convex, and $\boldsymbol{F}$ is continuous, there exists a fixed point solution for function $\boldsymbol{F}$, which corresponds to the best response strategies of price-setting data owners as it is obtained from Problem (28). According to Definition 4, we prove the existence of an NE. □

5.2 Uniqueness of the Nash Equilibrium

This subsection shows that a unique NE exists for Game PS-NCG. To obtain the NE, we determine the best response strategies for price-setting data owners through the following problem, where we substitute $x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$ given in (25) into the objective function (24a) of Problem (24).

$$p_w^{s^*} = \underset{p_w^s \geq 0}{\arg\max} \quad \left(p_w^s - \alpha_w^s\right)\left(x_w^{s^*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)\right)^2 \tag{29a}$$

$$\text{s.t.} \quad \eta_{x_w^s} \leq L_w^s\left(p_w^s, \boldsymbol{p}_{-w}^s\right) \leq \epsilon_{x_w^s}. \tag{29b}$$

Based on Lemma 4, we know that constraint (29b) is convex. We next prove that the objective function is concave, and thus, Problem (29) is a strictly concave maximization problem and admits a unique optimal solution. We present the result in Lemma 5 below.

Lemma 5. *There exists $p_w^{s,\max} \geq p_w^{s,\min}$, such that Problem* (29) *is a strictly concave maximization problem over the closed interval $\left[p_w^{s,\min}, p_w^{s,\max}\right]$, and the objective function is decreasing for $p_w^s > p_w^{s,\max}$. Thus, Problem* (29) *has a unique optimal solution over the same interval.*

Proof. To obtain the optimal solution of Problem (29), we set the first derivative of the objective function to zero, i.e., $dV_w^s/dp_w^s = 0$.

By substituting $dV_w^s/dp_w^s = 0$ into the second derivative of V_w^s, and given $d^2x_w^{s^*}/d\left(p_w^s\right)^2$, we have $d^2V_w^s/d\left(p_w^s\right)^2 \leq 0$. Therefore, V_w^s is concave at any stationary points, implying that at most one such point exists. We now consider the following two possible cases:

- There is one stationary point as denoted by $p_w^{s^*}$. Thus, an interval around $p_w^{s^*}$ exists where the objective function V_w^s is locally concave. Furthermore, V_w^s is decreasing for $p_w^s > p_w^{s,\max}$ due to the uniqueness of the stationary point.
- There is no stationary point, which implies that the objective function is either increasing or decreasing. If the objective function is strictly increasing, the optimal solution to Problem (29) is $p_w^{s,\max}$, and we have $p_w^{s,\min} = p_w^{s,\max}$. If the objective function is decreasing, V_w^s is concave in $\left[0, p_w^{s,\max}\right]$ and $p_w^{s,\min} = 0$.

□

We now prove the uniqueness of the NE in PS-NCG.

Theorem 5. *A unique NE exists in Game PS-NCG.*

Proof. The proof is based on the following lemma [29].

Lemma 6 ([29]). *A unique NE exists in PS-NCG if for all $w \in \mathcal{W}^s$*

- *The strategy space is a nonempty, convex, and compact subset of some Euclidean space.*
- *Player w's payoff V_w^s is continuous and strictly concave in p_w^s.*

Algorithm 1: $BR_w\left(\boldsymbol{p}^s_{-w}\right)$: Iterative Best Response Adaptation for a Price-setting Data Owner w

Initialize $i \leftarrow 0$, $p_w^{s^{(0)}}$, and ζ.
do
  The data owner submits the price strategy $p_w^{s^{(i)}}$ to the model requester.
  The data owner collects the model requester's responses $x_w^{s^*}\left(p^s_w, \boldsymbol{p}^s_{-w}\right)$ and $\frac{\partial x_w^{s^*}}{\partial p_w^s}$.
  The data owner updates its price strategy based on (30).
  $i \leftarrow i + 1$.
while $|p_w^{s^{(i)}} - p_w^{s^{(i-1)}}| > \zeta$;
Return $p_w^{s^{(i)}}$

Algorithm 2: Distributed Iterative Algorithm for Information Exchange among the Model Requester and Price-setting Data Owners

Initialize $k \leftarrow 0$, conver_flag $\leftarrow 0$, and ζ'.
while conver_flag = 0 **do**
  The model requester collects $p_w^{s^{(k)}}$ from all data owners $w \in \mathcal{W}^s$.
  Each data owner updates its best response strategy via Algorithm 1, i.e.,

$$p_w^{s^{(k+1)}} \leftarrow BR_w\left(\boldsymbol{p}^{s^{(k)}}_{-w}\right), \forall w \in \mathcal{W}^s$$

  The model requester checks the termination criterion:
  if $|p_w^{s^{(k+1)}} - p_w^{s^{(k)}}| < \zeta', \forall w \in \mathcal{W}^s$ **then**
    conver_flag $\leftarrow 1$.
  end
  $k \leftarrow k + 1$.
end
Return $\boldsymbol{p}^{s^{(k)}}$

According to Lemma 5, PS-NCG satisfies the above two properties, and we can conclude the uniqueness of the NE. □

5.3 Algorithm Design

In this subsection, we develop iterative algorithms for price-setting data owners and the model requester to obtain the unique NE of PS-NCG. We first develop an iterative algorithm for the price-setting data owners to update their best response strategies. We consider a price-setting data owner $w \in \mathcal{W}^s$, whose best response strategy can be obtained by solving Problem (29). Let $p_w^{s^{(i)}}$ and $x_w^{s^{*(i)}}$ denote the data owner w's and model requester's strategies at the i-th iteration. We update the data owner w's strategy using the following rule, which is obtained based on the gradient method:

$$p_w^{s^{(i+1)}} = \mathrm{Pro}\left(p_w^{s^{(i)}} + \eta^{(i)} x_w^{s^{*(i)}} \times \left[x_w^{s^{*(i)}} + 2\left(p_w^{s^{(i)}} - \alpha_w^s\right)\frac{\partial x_w^{s^*}}{\partial p_w^s}\Big|_{p_w^{s^{(i)}}}\right]\right), \tag{30}$$

where $\mathrm{Pro}(\cdot)$ denotes the projection onto the feasible region of Problem (29) and $\eta^{(i)}$ is the step size at iteration i.

We illustrate the proposed gradient-based algorithm in Algorithm 1. Price-setting data owner w first randomly initializes $p_w^{s(0)}$ (Step 1). In each iteration, the data owner submits its price to the model requester (Step 3). Meanwhile, the model requester computes $x_w^{s*(i)}$ and $\frac{\partial x_w^{s*}}{\partial p_w^s}\Big|_{p_w^{s(i)}}$ and announces them to the data owner. Upon reception of the model requester's responses (Step 4), the data owner updates the price based on (30) (Step 5). This procedure continues until convergence. The data owner checks the termination criterion (Step 7) and stops the algorithm when the relative changes of the prices during consecutive iterations are sufficiently small (as determined by the positive constant ζ). Utilizing this iterative algorithm, price-setting data owners only need to know the model requester's responses to their strategies. The proposed iterative algorithm has a relatively small communication overhead since only a few messages need to be exchanged.

Utilizing Algorithm 2, the model requester facilitates information exchange among the price-setting data owners. The model requester's response to each price-setting data owner reflects the strategies of other data owners (recall $x_w^{s*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$). As a result, each price-setting data owner does not need to know the strategies of others. In each iteration, the model requester shares the information the price-setting data owners need and receives their updated best response strategies. This procedure continues until convergence. We now prove the convergence of our algorithms. As stated in Lemma 5, Problem (29) is a strictly concave maximization problem. Therefore, the gradient-based Algorithm 1 converges to the optimal solution of this problem. To study the convergence of Algorithm 2, we follow the rationale of the proof in [44], which has been widely used in the literature (e.g., [30]). *Moreover, we would like to clarify that Algorithm 2 scales well in the number of (price-setting) data owners due to its parallel implementation, which will be empirically validated in Section 6.*

Theorem 6. *The proposed iterative algorithm shown in Algorithm 2 converges to the unique NE of PS-NCG.*

Proof. Algorithm 2 converges when two conditions are satisfied [44]: (1) Condition 1: A fixed point solution (i.e., NE) must exist; (2) Condition 2: Payoff function $V_w^s\left(x_w^{s*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right), p_w^s\right)$ is concave in $\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$. Theorem 5 states a unique NE exists for PS-NCG. Thus, Condition 1 is satisfied. Next, we verify Condition 2. Notice that according to [44], concavity is sufficient for the convergence while not necessary. Similar to Lemma 5, we now show that V_w^s is concave in $\left(p_w^s, \boldsymbol{p}_{-w}^s\right)$. According to the first-order condition for the concavity: $f\left(\boldsymbol{p}^s\right)$ is concave if for any $\boldsymbol{p}^{s'}$ and $\boldsymbol{p}^s$, we have

$$f\left(\boldsymbol{p}^{s'}\right) < f\left(\boldsymbol{p}^s\right) + \left(\boldsymbol{p}^{s'} - \boldsymbol{p}^s\right) \nabla f\left(\boldsymbol{p}^s\right).$$

For each price-setting data owner $w \in \mathcal{W}^s$, we need to show that

$$\left(p_w^{s'} - \alpha_w^s\right)\left(x_w^{s*}\left(p_w^{s'}, \boldsymbol{p}_{-w}^{s'}\right)\right)^2 < \left(p_w^s - \alpha_w^s\right)\left(x_w^{s*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)\right)^2 + \textstyle\sum_{m \in \mathcal{W}^s}\left(p_m^{s'} - p_m^s\right)\frac{\partial V_w^s}{\partial p_m^s}.$$

Lemma 5 shows V_w^s is concave in p_w^s. Therefore, to prove V_w^s is concave in $\boldsymbol{p}_{-w}^s$, we need show $\sum_{m \in \mathcal{W}^s \setminus \{w\}}\left(p_m^{s'} - p_m^s\right)\frac{\partial V_w^s}{\partial p_m^s} \geq 0$. Without loss of generality, we assume $\boldsymbol{p}^{s'} \geq \boldsymbol{p}^s$. If $\frac{\partial V_w^s}{\partial p_m^s} \geq 0$ for all $m \in \mathcal{W}^s \setminus \{w\}$, the above condition holds. We now derive $\frac{\partial V_w^s}{\partial p_m^s} = \frac{\partial x_w^{s*}}{\partial p_m^s}\left(2\left(p_w^s - \alpha_w^s\right)x_w^{s*}\left(p_w^s, \boldsymbol{p}_{-w}^s\right)\right)$. In the proof of Lemma 3, if $\lambda_w = 0$, $\psi_w = 0$, and $\beta > 0$, we get $p_w^s\left(x_w^{s*}\right)^3 = p_m^s\left(x_m^{s*}\right)^3, \forall m \in \mathcal{W}^s, m \neq w$, which implies that x_w^{s*} is non-decreasing in p_m^s. In this case, we have $\frac{\partial x_w^{s*}}{\partial p_m^s} \geq 0$, and thus $\frac{\partial V_w^s}{\partial p_m^s} \geq 0$. □

6 Numerical Results

6.1 Experimental Setup

We consider a data marketplace including a model requester, $N_s = 2$ price-setting data owners, and $N_t = 8$ price-taking data owners. Our experiments consist of two phases:

- Phase 1 focuses on solving the three-stage Stackelberg game and determining the allocation of privacy budget for each data owner in the DPFL-based model training task requested by the model requester. This process involves internal parameters, such as the privacy preferences of data owners, which vary across entities. However, directly obtaining these internal values from practical data marketplaces through surveys is challenging. Therefore, we utilize synthetic values for these parameters. Note that our developed three-stage Stackelberg game model and corresponding algorithms are applicable to real-world scenarios with different parameter values. Although the resulting prices and privacy budget may vary, the underlying principles remain unchanged.
- In phase 2, each data owner participates in the DPFL-based model training according to the privacy budget derived in phase 1. We use open data repositories (i.e., MNIST and CIFAR) and partition them across data owners to examine the impact of different solutions obtained in phase 1 on the model performance.

By default, the system parameters are configured as follows:

- The model requester's model quality constraint, denoted as S, is configured at 5. Notably, as per Theorem 2, a lower value of S corresponds to a higher model quality.
- The privacy preferences (α_w^s and α_w^t) of data owners are randomly sampled from a uniform distribution $\mathbb{U}(0.1, 1.0)$.
- The termination thresholds ζ and ζ' in Algorithms 1 and 2 are both fixed at 0.01. Besides, the step size $\eta^{(i)}$ used for price updates in (30) remains constant at 0.01.
- The benchmark parameter ψ_0 for characterizing the model requester's utility is chosen to be 200.

As mentioned in Section 1.3, none of the existing works considered the same problem setting as this paper. Hence, there is no suitable existing benchmark to compare with. Instead, we compare our derived SPE with the outcome of a DPFL-based data marketplace where the model requester only incentivizes price-taking data owners to participate in model training.

Furthermore, to assess the SPE's efficiency, we will conduct a comparative experiment between our proposed framework and the socially optimal solution. The efficiency of the SPE is quantified as the ratio between the social welfare achieved through the SPE and the maximum attainable social welfare. This ratio, bounded between 0 and 1, indicates how the market outcome deteriorates due to the self-interested behaviors exhibited by the involved players in the problem addressed in this paper.

The social welfare of the data marketplace is defined as the aggregate of payoffs for all players (including the model requester, price-setting data owners, and price-taking data owners). Mathematically, the social welfare is computed as

$$\Psi(\boldsymbol{x}) = \psi_0 - \sum_{w \in \mathcal{W}} 1/x_w - \sum_{w \in \mathcal{W}^s} \alpha_w^s (x_w^s)^2 - \sum_{w \in \mathcal{W}^t} \alpha_w^t (x_w^t)^2. \tag{31}$$

We denote by Ψ^* the maximum social welfare, and it can be determined by solving the following optimization problem:

$$\Psi^* \triangleq \max_{\boldsymbol{x} \in \mathcal{R}} \quad \Psi(\boldsymbol{x}) \tag{32a}$$

$$\text{s.t.} \quad \sum\nolimits_{w \in \mathcal{W}^t} \frac{1}{x_w^t} + \sum\nolimits_{w \in \mathcal{W}^s} \frac{1}{x_w^s} \leq S. \tag{32b}$$

Fig. 4. Price iterations as obtained by Algorithm 2 ($\zeta = \zeta' = \eta^{(i)} = 0.01$).

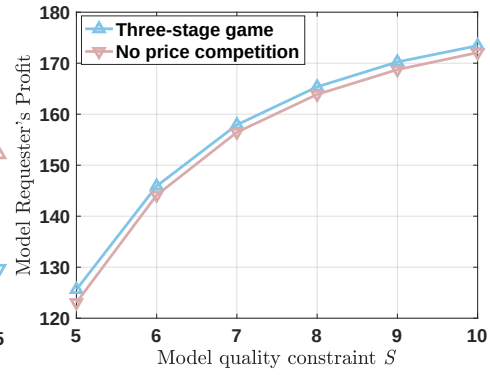

Fig. 5. The model requester's profit under different model quality constraints.

Here, $\mathcal{R}$ implies the feasible region of $\boldsymbol{x}$. Note that problem 32 is a convex optimization problem, and its unique optimal solution can be determined using a mathematical optimization solver known as CVX [11]. Then, if we denote the equilibrium strategies derived in our model as $\boldsymbol{x}^{NE}$, the SPE's efficiency can be defined as follows:

Definition 5 (SPE's Efficiency). *The SPE's efficiency is defined as the ratio between the social welfare obtained by the equilibrium and the maximum social welfare, i.e.,* $\Psi\left(\boldsymbol{x}^{NE}\right)/\Psi^*$.

6.2 Experimental Results

6.2.1 Convergence Behavior. We begin by examining the convergence rate of Algorithm 2. In each iteration, the price-setting data owners update and submit their best response strategies to the model requester. Figure 4 illustrates the price dynamics chosen by the price-setting data owners, which reveals a swift convergence of Algorithm 2. Theorem 6 states that this algorithm converges to the unique NE of Game PS-NCG. Therefore, we conclude that Algorithm 2 converges quickly to the SPE.

6.2.2 Model Requester's Profit. Figure 5 presents the profit obtained for varying values of the model quality constraint (i.e., S). We draw two key observations from Figure 5:

- First, it becomes less profitable for the model requester when it seeks to train a model of higher quality (i.e., a smaller value of S). This phenomenon occurs because the model requester must incentivize both price-setting and price-taking data owners to allocate more substantial privacy budgets to DPFL-based model training (as evidenced in Figure 6), resulting in higher payments.
- Second, the proposed three-stage game consistently outperforms the baseline scenario that lacks price-setting data owners (note that the total number of participating data owners remains constant). This improvement is attributed to the participation of price-setting data owners, which drives the prices for price-taking data owners down, as demonstrated later.

6.2.3 Allocated Privacy Budget. In Figure 6, we present the average privacy budget allocations made by data owners for DPFL-based model training with respect to various model quality constraints. The results yield two significant insights:

- First, it is evident that the model requester needs to incentivize data owners to allocate higher privacy budgets to DPFL-based model training when seeking to train a higher-quality model (which is characterized by a smaller value of S). This outcome is intuitive, as greater privacy budget allocation leads to weaker gradient obfuscation, resulting in improved model quality.

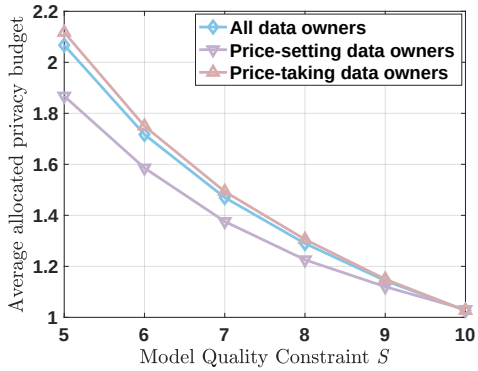

Fig. 6. Average privacy budget under different model quality constraints.

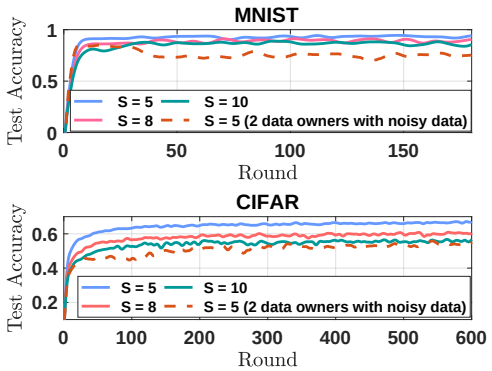

Fig. 7. Model testing accuracy under different model quality constraints.

- Second, we observe that price-taking data owners allocate higher privacy budgets than their price-setting counterparts. This disparity arises from the greater market power held by price-setting data owners, who are positioned to set higher prices. Consequently, the model requester will procure smaller privacy budgets from price-setting data owners.

6.2.4 Model Testing Accuracy. Figure 7 presents the model testing accuracy across various model quality constraints S. Our experiments involve collaboration between 8 price-taking and 2 price-setting data owners to respectively train the LeNet model [17] on the MNIST dataset and the CIFAR-CNN model [35] on the CIFAR dataset for the model requester. The training lasts $R = 180(600)$ rounds, with the datasets uniformly and randomly allocated to each data owner. In each round, the data owners perform 20(30) epochs of local training using the mini-batch stochastic gradient descent algorithm with a batch size of 256(64) and a learning rate of 0.1. The Gaussian noise variance for gradient perturbation is determined based on Lemma 2 given data owners' allocated privacy budget (computed using (20) and (25)), along with other tuned system parameters. Figure 7 demonstrates that increasing the model quality constraint S leads to a decrease in model accuracy due to the reduction in the privacy budget, as depicted in Figure 6.

6.2.5 Social Efficiency. Figure 8 shows the ratio between the social welfare obtained at an SPE and the maximum social welfare. We have two-fold observations on Figure 8:

- First, as the number of price-setting data owners increases (consequently leading to a decrease in the number of price-taking data owners while keeping the total number of data owners constant), the efficiency ratio decreases. This trend occurs due to the increased influence of price-setting data owners in market dynamics, potentially manipulating the outcome.
- Second, the efficiency of the SPE declines as the model requester's model quality requirement becomes more stringent (i.e., a smaller value of S). This phenomenon emerges because the model requester must acquire more privacy budgets from (price-setting) data owners, who can manipulate the market and thereby render the SPE less socially efficient.

6.2.6 Data Owners' Payoff. Figure 9 illustrates the average payoffs obtained by price-setting and price-taking data owners. We have two-fold observations on this:

- First, a higher requirement for model quality, denoted by smaller values of S, results in larger payoffs for data owners.
- In terms of a comparison between the payoffs of price-setting and price-taking data owners: When a relatively higher-quality model is demanded (i.e., with a smaller S), price-setting data

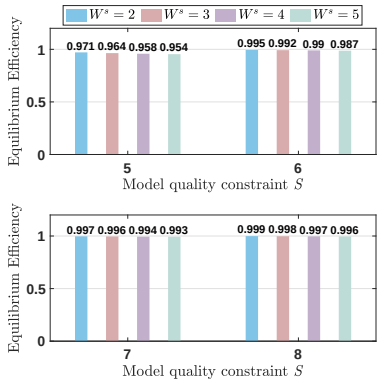

Fig. 8. Social efficiency of equilibrium.

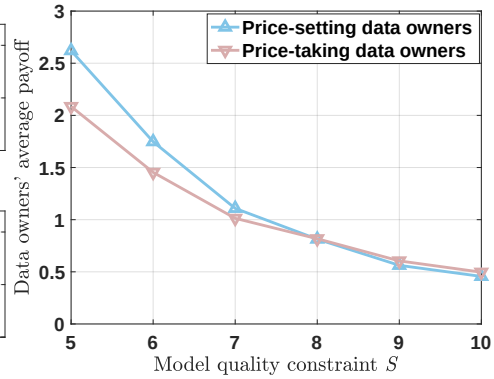

Fig. 9. Average payoffs of data owners.

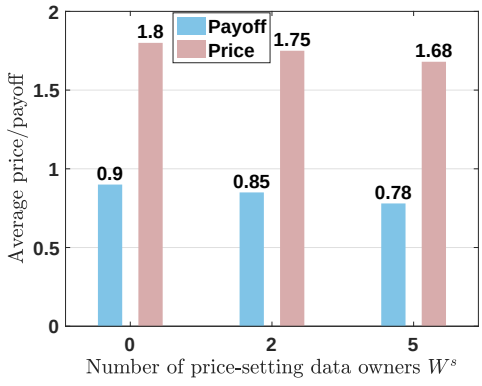

Fig. 10. Average payoff and optimal price $\boldsymbol{p}_t$ of price-taking data owners ($W = 10$ and $S = 10$).

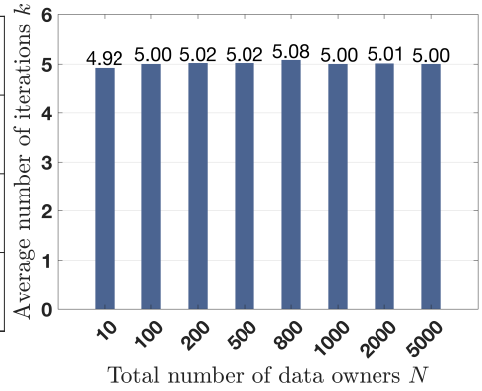

Fig. 11. Average number of iterations of Algorithm 2 under different number of (price-setting) data owners.

owners tend to achieve higher average payoffs than price-taking data owners. Conversely, when a relatively lower-quality model is sought (i.e., with a larger S), price-taking data owners tend to attain higher average payoffs than their price-setting counterparts. These outcomes are attributed to the variations in allocated privacy budgets under different model quality requirements and the distinct payment functions employed for price-setting (quadratic) and price-taking (linear) data owners.

6.2.7 Payoff and Price of Price-taking Data Owners. We further investigate the impact of price competition among price-setting data owners on the market prices established by the model requester for price-taking data owners (i.e., denoted as $\boldsymbol{p}_t$). In this analysis, we specifically sample the privacy preferences of data owners from a uniform distribution $\mathbb{U}(0.8, 1.0)$ to accentuate the dynamics in price-taking data owners' prices and payoffs. Figure 10 depicts the average payoffs of price-taking data owners and the corresponding prices set by the model requester when different numbers of price-setting data owners participate in the market. From this figure, we can observe that the involvement of price-setting data owners exerts downward pressure on market prices, given their increased market power. Consequently, price-taking data owners experience a reduction in their payoffs.

6.2.8 Scalability of Proposed Algorithms. Finally, we examine the scalability of our proposed iterative algorithms (i.e., Algorithm 2) for price updates of price-setting data owners. Figure 11 displays the average number of iterations versus the total number of data owners, with 20% of them being price-setting ones. Each data point in the figure corresponds to the average of 50 independent simulation runs. The results clearly demonstrate that the average number of iterations required for Algorithm 2 to converge remains nearly constant regardless of the number of (price-setting) data owners. This observation validates the desirable scalability of the algorithm with respect to the market size.

7 Discussions

- ***Data Owners with Noisy Data***: In practice, data owners may possess noisy data, leading to inaccurate computed stochastic gradient and compromised model accuracy. This is demonstrated by the dashed lines in Figure 7, where 2 data owners possess randomly generated data (achieved via label shuffling). This greatly reduces the contributions of data owners with noisy data to model training, which can result in unfair payments if only privacy costs are considered, as in this work. To address this, we can incorporate data owner contribution evaluation mechanisms, such as the Shapley value [23], to accurately assess their contributions and determine appropriate payments. We will consider integrating contribution evaluation and privacy compensation in the future.
- ***Arbitrage-free Pricing***: Our data marketplace design poses an arbitrage risk. A model requester could perform two independent model training sessions, each requiring a smaller privacy budget, to train two lower-quality models at a lower cost. By using ensemble learning techniques to combine these models and obtain a higher-quality model, while paying less than the cost of directly training the higher-quality model, arbitrage can occur. In future work, we will explore an arbitrage-free pricing design to address this issue.

8 Conclusion

This paper presented a novel data marketplace based on differentially private federated learning, which facilitates interactions between the model requester and both price-setting and price-taking data owners. To analyze these interactions, we adopted a three-stage Stackelberg game framework and focused on maximizing the model requester's profit. Through our analysis, we demonstrated that the formulated game is a convex game with a unique subgame perfect equilibrium. To determine the equilibrium strategies, we developed iterative algorithms for the model requester and price-setting data owners. Our experimental results revealed that our proposed three-stage framework outperforms a baseline scenario that does not involve price-setting data owners in terms of the model requester's profit. Besides, the results indicated that price competition among price-setting data owners reduces market prices compared to a situation where all data owners are price-taking.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62102337, 62202307, 62122066, 62102352, 62372163, 62202150, and U23A20306), the Natural Science Foundation of Hunan Province, China (No. 2023JJ40174), the Young Elite Scientists Sponsorship Program by CAST (No. 2023QNRC001), the Fundamental Research Funds for the Central Universities, the National Key Research and Development Program of China (No. 2021YFB3101100), the Hunan Provincial Key Research and Development Program (No. 2024AQ2041 and 2022NK2046), the Yuelushan Industrial Innovation Center Cultivation Project (No. 2023YC110130), the Zhejiang Province Pioneer Plan (No. 2024C01074).

References

[1] Anish Agarwal, Munther Dahleh, and Tuhin Sarkar. 2019. A marketplace for data: An algorithmic solution. In *Proceedings of ACM Conference on Economics and Computation*.

[2] Léon Bottou, Frank E Curtis, and Jorge Nocedal. 2018. Optimization methods for large-scale machine learning. *SIAM Rev.* 60, 2 (2018), 223–311.

[3] Mark Bun and Thomas Steinke. 2016. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Proceedings of Theory of Cryptography Conference*.

[4] Raul Castro Fernandez. 2023. Data-Sharing Markets: Model, Protocol, and Algorithms to Incentivize the Formation of Data-Sharing Consortia. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–25.

[5] Junjie Chen, Minming Li, and Haifeng Xu. 2022. Selling Data To a Machine Learner: Pricing via Costly Signaling. In *International Conference on Machine Learning*. PMLR, 3336–3359.

[6] Lingjiao Chen, Paraschos Koutris, and Arun Kumar. 2019. Towards model-based pricing for machine learning in a data marketplace. In *Proceedings of International Conference on Management of Data*.

[7] DEFINED.AI. [n. d.]. https://www.defined.ai/.

[8] 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.

[9] Hao Fang, Bin Chen, Xuan Wang, Zhi Wang, and Shu-Tao Xia. 2023. GIFD: A Generative Gradient Inversion Method with Feature Domain Optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4967–4976.

[10] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Gong. 2020. Local model poisoning attacks to byzantine-robust federated learning. In *Proceedings of {USENIX} Security Symposium*.

[11] M. Grant and S. Boyd. 2014. CVX: Matlab software for disciplined convex programming, version 2.1. *http://cvxr.com/cvx* (2014).

[12] Rui Hu and Yanmin Gong. 2020. Trading data for learning: Incentive mechanism for on-device federated learning. In *Proceedings of IEEE Global Communications Conference*.

[13] Chao Huang, Ming Tang, Qian Ma, Jianwei Huang, and Xin Liu. 2023. Promoting Collaborations in Cross-Silo Federated Learning: Challenges and Opportunities. *IEEE Communications Magazine* (2023).

[14] Innodata. [n. d.]. https://innodata.com/ai-data-marketplace/.

[15] Prateek Jain, Purushottam Kar, et al. 2017. Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning* 10, 3-4 (2017), 142–363.

[16] Minchul Kim, Anil K Jain, and Xiaoming Liu. 2022. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18750–18759.

[17] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. 1989. Handwritten digit recognition with a back-propagation network. *Proceedings of Advances in Neural Information Processing Systems* 2 (1989).

[18] Zitao Li, Bolin Ding, Ce Zhang, Ninghui Li, and Jingren Zhou. 2021. Federated matrix factorization with privacy guarantee. *Proceedings of the VLDB Endowment* 15, 4 (2021).

[19] Zhuohang Li, Jiaxin Zhang, Luyang Liu, and Jian Liu. 2022. Auditing privacy defenses in federated learning via generative gradient leakage. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10132–10142.

[20] Jinfei Liu, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. 2021. Dealer: An end-to-end model marketplace with differential privacy. *Proceedings of the VLDB Endowment* 14, 6 (2021), 957–969.

[21] Junxu Liu, Jian Lou, Li Xiong, Jinfei Liu, and Xiaofeng Meng. 2021. Projected federated averaging with heterogeneous differential privacy. *Proceedings of the VLDB Endowment* 15, 4 (2021), 828–840.

[22] Ken Liu, Shengyuan Hu, Steven Z Wu, and Virginia Smith. 2022. On privacy and personalization in cross-silo federated learning. *Advances in Neural Information Processing Systems* 35 (2022), 5925–5940.

[23] Zelei Liu, Yuanyuan Chen, Han Yu, Yang Liu, and Lizhen Cui. 2022. GTG-Shapley: Efficient and accurate participant contribution evaluation in federated learning. *ACM Transactions on Intelligent Systems and Technology (TIST)* 13, 4 (2022), 1–21.

[24] Xu Ma, Xiaoqian Sun, Yuduo Wu, Zheli Liu, Xiaofeng Chen, and Changyu Dong. 2022. Differentially private byzantine-robust federated learning. *IEEE Transactions on Parallel and Distributed Systems* 33, 12 (2022), 3690–3701.

[25] Bernard Marr. 2017. Where can you buy big data? here are the biggest consumer data brokers. *Forbes* 1 (2017).

[26] Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. 1995. *Microeconomic Theory*. Oxford University Press, New York.

[27] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of Artificial Intelligence and Statistics*.

[28] Jian Pei. 2020. A survey on data pricing: from economics to data science. *IEEE Transactions on knowledge and Data Engineering* 34, 10 (2020), 4586–4608.
[29] J. B. Rosen. 1965. Existence and uniqueness of equilibrium points for concave N-person games. *Econometrica* 33, 3 (1965), 347–351.
[30] Hamed Shah-Mansouri, Vincent WS Wong, and Jianwei Huang. 2017. An incentive framework for mobile data offloading market under price competition. *IEEE Transactions on Mobile Computing* 16, 11 (2017), 2983–2999.
[31] Yoav Shoham and Kevin Leyton-Brown. 2008. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.
[32] Peng Sun, Haoxuan Che, Zhibo Wang, Yuwei Wang, Tao Wang, Liantao Wu, and Huajie Shao. 2021. Pain-FL: Personalized Privacy-Preserving Incentive for Federated Learning. *IEEE Journal on Selected Areas in Communications* 39, 12 (2021), 3805–3820.
[33] Peng Sun, Xu Chen, Guocheng Liao, and Jianwei Huang. 2022. A Profit-Maximizing Model Marketplace with Differentially Private Federated Learning. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 1439–1448.
[34] Qiheng Sun, Xiang Li, Jiayao Zhang, Li Xiong, Weiran Liu, Jinfei Liu, Zhan Qin, and Kui Ren. 2023. Shapleyfl: Robust federated learning based on Shapley value. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2096–2108.
[35] Yanan Sun, Xian Sun, Yuhan Fang, Gary G Yen, and Yuqiao Liu. 2021. A novel training protocol for performance predictors of evolutionary neural architecture search algorithms. *IEEE Transactions on Evolutionary Computation* 25, 3 (2021), 524–536.
[36] Hideaki Takahashi, Jingjing Liu, and Yang Liu. 2023. Breaching FedMD: Image Recovery via Paired-Logits Inversion Attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12198–12207.
[37] Xin Wang, Jianping He, Peng Cheng, and Jiming Chen. 2018. Privacy preserving collaborative computing: Heterogeneous privacy guarantee and efficient incentive mechanism. *IEEE Transactions on Signal Processing* 67, 1 (2018), 221–233.
[38] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2019. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *Proceedings of IEEE International Conference on Computer Communications*.
[39] Rachel Wilka, Rachel Landy, and Scott A McKinney. 2017. How machines learn: where do companies get data for machine learning and what licenses do they need. *Wash. JL Tech. & Arts* 13 (2017), 217.
[40] Zihang Xiang, Tianhao Wang, Wanyu Lin, and Di Wang. 2023. Practical Differentially Private and Byzantine-resilient Federated Learning. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–26.
[41] Cong Xie, Sanmi Koyejo, and Indranil Gupta. 2020. Zeno++: Robust fully asynchronous SGD. In *International Conference on Machine Learning*. PMLR, 10495–10503.
[42] Anran Xu, Zhenzhe Zheng, Fan Wu, and Guihai Chen. 2022. Online Data Valuation and Pricing for Machine Learning Tasks in Mobile Health. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 850–859.
[43] Yuchen Yang, Bo Hui, Haolin Yuan, Neil Gong, and Yinzhi Cao. 2023. {PrivateFL}: Accurate, Differentially Private Federated Learning via Personalized Data Transformation. In *32nd USENIX Security Symposium (USENIX Security 23)*. 1595–1612.
[44] R. D. Yates. 1995. A framework for uplink power control in cellular radio systems. *IEEE Journal on Selected Areas in Communications* 13, 7 (1995), 1341–1347.
[45] Zhenning Yi, Yutao Jiao, Wenting Dai, Guoxin Li, Haichao Wang, and Yuhua Xu. 2022. A Stackelberg Incentive Mechanism for Wireless Federated Learning With Differential Privacy. *IEEE Wireless Communications Letters* 11, 9 (2022), 1805–1809.
[46] Jinliang Yuan, Shangguang Wang, Shihe Wang, Yuanchun Li, Xiao Ma, Ao Zhou, and Mengwei Xu. 2023. Privacy as a Resource in Differentially Private Federated Learning. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*. IEEE, 1–10.
[47] Kai Yue, Richeng Jin, Chau-Wai Wong, Dror Baron, and Huaiyu Dai. 2023. Gradient obfuscation gives a false sense of security in federated learning. In *32nd USENIX Security Symposium (USENIX Security 23)*. 6381–6398.
[48] Zaloni. [n. d.]. https://www.zaloni.com/next-generation-data-marketplace/.
[49] Lefeng Zhang, Tianqing Zhu, Ping Xiong, Wanlei Zhou, and S Yu Philip. 2022. A robust game-theoretical federated learning framework with joint differential privacy. *IEEE Transactions on Knowledge and Data Engineering* 35, 4 (2022), 3333–3346.
[50] Yang Zhang and Ying-Ju Chen. 2020. Optimal nonlinear pricing in social networks under asymmetric network information. *Operations Research* 68, 3 (2020), 818–833.
[51] Shuyuan Zheng, Yang Cao, and Masatoshi Yoshikawa. 2023. Secure Shapley value for cross-silo federated learning. *Proceedings of the VLDB Endowment* 16, 7 (2023), 1657–1670.

[52] Shuyuan Zheng, Yang Cao, Masatoshi Yoshikawa, Huizhong Li, and Qiang Yan. 2022. FL-Market: Trading private models in federated learning. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 1525–1534.
[53] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. In *Proceedings of Advances in Neural Information Processing Systems*.

Received 17 January 2024; revised 25 April 2024; accepted 6 May 2024