From Data Collection to Learning from Distributed Data: a Minimum Cost Incentive Mechanism for Private Discrete Distribution Estimation and an Optimal Stopping Approach for Iterative Training in Federated Learning

by

Pengfei Jiang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved May 2020 by the
Graduate Supervisory Committee:

Lei Ying, Chair
Junshan Zhang
Yanchao Zhang
Weina Wang

ARIZONA STATE UNIVERSITY

December 2020

ABSTRACT

The first half of this dissertation introduces a minimum cost incentive mechanism for collecting discrete distributed private data for big-data analysis. The goal of an incentive mechanism is to incentivize informative reports and make sure randomization in the reported data does not exceed a target level. It answers two fundamental questions: *what is the minimum payment required to incentivize an individual to submit data with quality level ϵ?* and *what incentive mechanisms can achieve the minimum payment?* A lower bound on the minimum amount of payment required for guaranteeing quality level ϵ is derived. Inspired by the lower bound, our incentive mechanism (WINTALL) first decides a winning answer based on reported data, then pays to individuals whose reported data match the winning answer. The expected payment of WINTALL matches lower bound asymptotically. Real-world experiments on Amazon Mechanical Turk are presented to further illustrate novelty of the principle behind WINTALL.

The second half studies problem of iterative training in Federated Learning. A system with a single parameter server and M client devices is considered for training a predictive learning model with distributed data. The clients communicate with the parameter server using a common wireless channel so each time, only one device can transmit. The training is an iterative process consisting of multiple rounds. Adaptive training is considered where the parameter server decides when to stop/restart a new round, so the problem is formulated as an optimal stopping problem. While this optimal stopping problem is difficult to solve, a modified optimal stopping problem is proposed. Then a low complexity algorithm is introduced to solve the modified problem, which also works for the original problem. Experiments on a real data set shows significant improvements compared with policies collecting a fixed number of updates in each iteration.

To my family.

ACKNOWLEDGMENTS

First and foremost, I would like to express my gratitude to my phenomenal advisor, Professor Lei Ying for his guidance, patience, and encouragement. His enthusiasm, knowledge, and rigorous attitude toward research have deeply influenced me. He has always been a great mentor who dedicated numerous amount of time and tremendous efforts in my doctoral study. I am eternally grateful to have him as my advisor who made my Ph.D. journey rewarding.

I would like to thank Professors Junshan Zhang, Yanchao Zhang and Weina Wang for serving on my committee, and for their valuable insights and feedback on my thesis.

The past several years would not have been as enjoyable without the company of terrific colleagues at INLAB and friends. Thank you for all of understanding, help and joys brought to me in this long journey.

Finally, I would like to thank my parents for their unconditional support and love. And I would like to extend my sincere thanks to my wife Haiyan Sun for her love, company, encouragement and support during my Ph.D. journey.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

Chapter 1

INTRODUCTION

This dissertation focuses on two related topics in machine learning field: (1) designing incentive mechanism for collecting private data, (2) designing algorithm for distributed learning.

1.1 Background and Introduction on Design of Incentive Mechanisms for Private Discrete Distribution Estimation

The access to massive personal data via online/offline platforms enables new scientific discoveries, new personalized applications and services, and new machine learning algorithms. The success of big-data-based applications/services require the participation of, or access to, a massive population. In many systems, this massive participation is a result of a highly popular application or service, such as Gmail, iPhone or Amazon, which attracts millions of active users. In other systems such as SurveyMonkey or Amazon Mechanical Turk, the massive participation is achieved using monetary incentives.

This dissertation focuses on the design of incentive mechanisms for attracting data subjects with monetary incentives instead of services. We consider a big-data system where the platform elicits personal data from a crowd. The design of efficient incentive mechanisms differs from other systems in the following aspects.

- The quality of the reported data is "controlled" by an individual, and often "unverifiable" by the platform. This is because the platform does not have direct access to true personal data so it will not be able to verify the truthfulness of the reports. Furthermore, when the privacy-preserving mechanism is controlled by

individuals, the amount of randomization added to the data is also unverifiable to the platform. Therefore, the conventional wisdom of "pay according to the quality" is difficult to implement.

- Many incentive mechanisms aim at "truthfulness" to make sure an individual has no incentive to alter her/his answer and reports the true data. This however is not necessary and sometimes should be avoided in collecting personal information. Companies, such as Google and Apple, increasingly emphasize privacy protection of their customers and prefer privacy-preserving data instead of raw personal information. Both Google and Apple have pioneered in using differential privacy in data collection Erlingsson *et al.* (2014); Apple (2020)

In light of the challenges above, this dissertation studies the design of incentive mechanisms aiming at obtaining private data with target quality with minimum payment, for which we need to understand *who should be paid* and *how much should be paid*? Both questions are highly nontrivial because a platform has limited information for assessing the quality of reported data. A flat-rate-payment mechanism, which gives a predetermined payment to each data subject, is not cost-efficient. This is because rational individuals, who are interested in maximizing their payoffs (payoff = payment − cost), would not want to leak any personal information. So they will submit completely random answers under a flat-rate-payment mechanism.

This dissertation considers a model where a platform is interested in collecting private data from N individuals for discrete distribution estimation. The private data of individual i is denoted by S_i, which is a sample drawn from an underlying discrete distribution θ. An individual reports X_i, where the conditional distribution of X_i given S_i is controlled by user i via ϵ_i, the privacy budget (i.e., privacy loss) of individual i. An example of this model is presented in Figure 1.1.

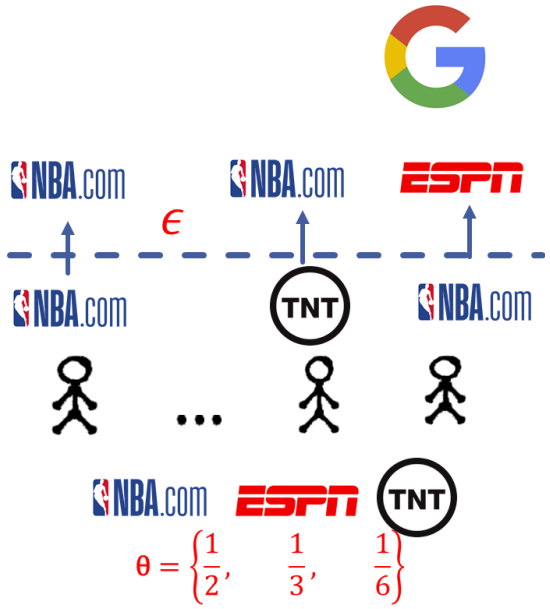

Figure 1.1: Example of Platform (Google) Collecting Web Browsing Information from Individuals

In this example, the platform (Google) is collecting web browsing information from individuals. Consider three websites that have information about NBA finals: NBA, ESPN and TNT. The distribution that these three websites are visited is $\theta = \left(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right)$, which the platform wants to learn. When the web browsing information is reported to the platform, the answer is randomized to protect user privacy. For example, the report from the rightmost user was changed from NBA, which is the private information S_i, to ESPN, which is the reported information X_i. An individual can change the privacy budget ϵ to adjust the level of randomization. Given this model, we first derive a lower bound on the minimum payment required given a target quality level, and then propose an incentive mechanism that matches the lower bound under some assumptions.

1.2 Related Work on Design of Incentive Mechanisms for Private Data Collection

Most existing work on privacy-aware data collection Ghosh and Roth (2011); Fleischer and Lyu (2012); Ligett and Roth (2012); Roth and Schoenebeck (2012); Ghosh and Ligett (2013); Nissim *et al.* (2014); Ghosh *et al.* (2014) assumes that the data collector is trustworthy and the privacy protection is implemented by the *data collector* when the data collector releases the data. Therefore, the goal is to incentivize truthful reporting from data subjects. This dissertation considers a fundamentally different model where the data collector is not necessarily trustworthy. Privacy is embedded in the reporting and the privacy budget is controlled directly by data subjects. From the best of our knowledge, incentive mechanisms for such a model have been studied only very recently. The most closely related line of work is Wang *et al.* (2015, 2016), which studied the market value of private data by casting the problem as eliciting private data from privacy-sensitive individuals. Both papers consider binary data, which is a special case of the model studied in this dissertation. Cai *et al.* (2015) investigated a similar problem under a model, where the reported data is the true answer plus an additive noise with mean zero and variance as a function of effort level ϵ. This dissertation does not assume an additive noise model. In fact, noises introduced by many popular privacy protection mechanisms based on differential privacy are not additive.

Another line of research closely related to the problem studied in this dissertation is the design of incentive mechanisms for crowdsourcing to obtain high-quality answers without knowing the ground truth. A popular approach used in crowdsourcing is the peer prediction mechanisms Prelec (2004); Von Ahn and Dabbish (2004); Miller *et al.* (2009); Witkowski and Parkes (2012); Radanovic and Faltings (2013, 2014, 2015); Dasgupta and Ghosh (2013); Shnayder *et al.* (2016), under which each

individual is paired with another randomly selected individual and is paid based on how well her reported data predicts the data from her paired individual. Von Ahn and Dabbish (2004) proposed an output agreement mechanism where a positive payment is made if two answers agree. Prelec (2004) introduced the "Bayesian Truth Serum" (BTS) mechanism, which requires a data subject to provide her own answers as well as her belief of others' answers. A high score is given to an answer when the actual frequency is larger than the prediction. The mechanism has been further extended in various different settings Witkowski and Parkes (2012); Radanovic and Faltings (2013, 2014, 2015). Dasgupta and Ghosh (2013); Shnayder *et al.* (2016) introduced strong truthfulness mechanisms for binary and non-binary signals in the presence of multiple questions. Shah and Zhou (2015); Shah *et al.* (2015); Shah and Zhou (2016) developed incentive mechanisms for improving quality of labelling in crowdsourcing. The mechanisms incentivize workers to self-correct their answers in a second stage after comparing their answers with a reference answer from other workers. The goal of these mechanisms is to obtain "truthful" answers, while under our model, the platform is interested in eliciting answers with target quality instead of truthful answers. Gong and Shroff (2018) developed a truthful crowdsourcing incentive mechanisms for quality, effort and data elicitation. The truthful incentive mechanism aims at maximizing social welfare instead of minimizing the payment of the data collector, which also differs from this dissertation.

Khetan and Oh (2016) studies the problem of maximizing accuracy of crowdsouced data given a fixed budget, under which crowdsoucing tasks are assigned adaptively based on the answers collected. The dissertation introduced an adaptive mechanism combined with inference scheme. Wauthier and Jordan (2011) introduced a Bayesian inference model for crowdsourcing which integrates data collection and learning. The focus of Wauthier and Jordan (2011); Khetan and Oh (2016) is on task assignment

instead of incentive mechanisms. Liu and Liu (2015) developed a learning algorithm to identify low- and high-quality labelers and further used this information to improve the labelling quality in crowdsourcing. In Cummings *et al.* (2015), the authors considered a model under which the data collector can buy data with different variance levels with different prices. The focus is on incentivizing data providers to report their true cost functions, and it assumes the variance levels tagged are the true variance levels of the data, which is fundamental different from our model where the variance level is unverifiable to the data collector.

1.3 Background and Introduction on Federated Learning

Most existing machine learning applications for big-data analytics require the models to be trained in data centers, which raises significant privacy concerns when data used contain sensitive personal information such as clicks, photos, etc. So second part of this dessertation considers problem of machine learning on distributed data. Federated learning is a distributed machine learning framework proposed by Google [1] to train a machine learning model with datasets distributed over local devices (such as mobile phones) instead of in data centers.

Training process is run on distributed device such as mobile phones so that a device does not need to expose personal data on the device to servers or other devices. The updates for the model (e.g. the gradients of SGD) will be transmitted to a parameter server which will aggregate the updates to update the machine learning model. The updated model will then be broadcast to the devices for the next iteration of training.

[1] `https://ai.googleblog.com/2017/04/federated-learning-collaborative.html`

1.4 Related Work on Federated Learning

Federated learning has applications in many areas Yang *et al.* (2019), e.g. Google has implemented federated learning in their Gboard Chen *et al.* (2019); Hard *et al.* (2018), where a neural network language model is trained using data on personal mobile devices for next-word prediction. Due to randomness and uncertain in data processing and transmissions, it has been observed that even with dedicated servers, learning can be slowed down significantly by a few machines that take unusually long time to complete the training. The problem becomes even worse in Federated Learning where devices have heterogeneous capacities, and are less reliable. Therefore, a critical problem in Federated Learning is to schedule the training, in particular for those algorithms that require iterative training. In the past, Wang and Joshi (2018a,b) have studied the convergence of the loss function with respect to the number of local iterations on each client and proposed mechanisms to optimally select the number of local iterations on each client. This dissertation considers a different problem and considers when the parameter server should stops the current iteration, updates the machine learning model and starts the next iteration. Such a decision is based on the number of updates received, the expected waiting time to receive the next update, and how the loss function decreases as the number of updates increases.

1.5 Summary of Contributions

Contributions of this dissertation are composed of two related parts. In Chapter 2, we studied incentive mechanisms for private discrete distribution estimation. We first derived a lower bound on the minimum payment required for guaranteeing quality level, and then proposed WINTALL — a novel incentive mechanism. The expected payment of WINTALL matches the lower bound when the underlying parameter θ

can be estimated by the platform accurately. We present its application to private discrete distribution estimation, where WINTALL rewards individuals whose reported answers match the most popular one. We also presented real-world experiments on Amazon Mechanical Turk to validate the novelty of WINTALL-inspired mechanisms.

In Chapter 3, we studied the problem of iterative training in Federated Learning. We consider a system with a single parameter server (PS) and M client devices for training a predictive learning model with distributed data sets on the client devices. The clients communicate with the parameter server using a common wireless channel so each time, only one device can transmit. The training is an iterative process consisting of multiple rounds. At beginning of each round, each client trains the model, broadcast by the parameter server at the beginning of the round, with its own data. After finishing training, the device transmits the update to the parameter server when the wireless channel is available. The server aggregates updates to obtain a new model and broadcasts it to all clients to start a new round. We consider adaptive training where the parameter server decides when to stop/restart a new round, and formulate the problem as an optimal stopping problem. While this optimal stopping problem is difficult to solve, we propose a modified optimal stopping problem. We first develop a low complexity algorithm to solve the modified problem, which also works for the original problem. Experiments on a real data set shows significant improvements compared with policies collecting a fixed number of updates in each iteration. the problem as an optimal stopping problem and develop a low complexity algorithm to solve the stopping rule. Our experiments on real dataset shows the significant improvements compared with policies that collect a fixed number of updates in each iteration.

Chapter 2

A MINIMUM COST INCENTIVE MECHANISM FOR PRIVATE DISCRETE DISTRIBUTION ESTIMATION

This chapter studies the design of incentive mechanisms aiming at obtaining private data with target quality with minimum payment, for which we need to understand *who should be paid* and *how much should be paid*? Both questions are highly nontrivial because a platform has limited information for assessing the quality of reported data. A flat-rate-payment mechanism, which gives a predetermined payment to each data subject, is not cost-efficient. This is because rational individuals, who are interested in maximizing their payoffs (payoff = payment − cost), would not want to leak any personal information. So they will submit completely random answers under a flat-rate-payment mechanism.

This chapter considers a model where a platform is interested in collecting private data from N individuals for discrete distribution estimation. The private data of individual i is denoted by S_i, which is a sample drawn from an underlying discrete distribution θ. An individual reports X_i, where the conditional distribution of X_i given S_i is controlled by user i via ϵ_i, the privacy budget (i.e., privacy loss) of individual i. Given this model, we first derive a lower bound on the minimum payment required given a target quality level, and then propose an incentive mechanism that matches the lower bound under some assumptions.

2.1 Main Results

We formulate the design of minimum payment incentive mechanisms as an optimization problem in Section 2.2 assuming a cost-aware platform who is interested

in minimizing total payment when eliciting private data with required quality, and strategic individuals who are interested in maximizing their payoffs. In Section 2.3, we derive a lower bound on the minimum payment. The lower bound is derived by introducing a genie-aided mechanism, where a genie knows θ. Under the genie-aided mechanism, the platform pays an individual based on only her reported answer and θ, so the minimum payment problem can be decoupled to minimum payment problems for each individual. We further show that any incentive mechanism that does not have access to θ can be mimicked by the genie-aided mechanism with the same expected payment. Therefore, the minimum payment under the genie-aided mechanism is a lower bound on the original problem.

Inspired by the lower bound, we propose WINTALL, a winners-take-all incentive mechanism in Section 2.3. WINTALL decides the winning value based on the reported data, the privacy protection mechanism, and the target privacy budget ϵ, and pays to individuals whose reported data match the winning value. The payment is determined based on the marginal privacy cost of the individuals, the privacy protection mechanism and the target privacy budget. For example, under the k-ary randomized response mechanism Kairouz *et al.* (2016), the winning value turns out to be the most popular value among the reported data, and the amount of payment is

$$g'(\epsilon)\frac{(M+\epsilon)\epsilon}{M\frac{\sum_{i=1}^{N}1_{x_i=m^*}}{N}-1}, \tag{2.1}$$

where M is the size of the alphabet (sample space), m^* is the winning value, and $g(\epsilon)$ is the cost of reporting data with privacy budget ϵ.

2.2 Model

Let i be the index of individuals and $S_i \in \mathcal{S}$ be the private data of individual i, where $\mathcal{S}$ is a finite set. We assume $S_i \in \mathcal{S}$ to be a discrete random variable with

distribution θ, and assume $\{S_i\}$ are independent across individuals. We remark when individuals are chosen uniformly at random from a large population like in most online or offline surveys, it can be viewed as sampling without replacement. Given θ, the independent assumption holds under sampling without replacement.

Let $X_i \in \mathcal{X}$ denote the data that individual i reports to the platform, where $\mathcal{X}$ is a finite set and may be different from $\mathcal{S}$. Furthermore, denote by $\sigma_i(\epsilon) : \mathcal{S} \to \mathcal{X}$ a data-reporting mechanism that generates reported data X_i according to the private data with privacy budget ϵ. X_i is also a discrete random variable. For example, Google RAPPOR and Apple iPhone have implemented privacy-preserving mechanisms based on differential privacy Erlingsson *et al.* (2014); Apple (2020) where ϵ is the privacy budget defined in differential privacy.

We assume that the privacy budget ϵ and the privacy preserving mechanism uniquely determine the distribution

$$\mathbb{P}_{X|S}\left(x|s;\epsilon\right). \tag{2.2}$$

We assume a common privacy preserving mechanism is used by all individuals but the privacy budget ϵ_i is controlled by individual i. We further assume $\mathbb{P}_{X|S}\left(x|s;\epsilon\right)$ is differentiable with respect to ϵ. Note that this assumption means an individual controls $\mathbb{P}_{X_i|S_i}$ via privacy budget instead of dictating every bit of reported data. We believe this is a realistic assumption. For example, we can easily envision that Google or Apple in the future may let users determine the level of privacy-privacy they prefer and allow the individuals to set the value in their browser or iPhone settings, but it is difficult to image that an individual would have the time and knowledge to customize every single bit of her personal data reported to Google or Apple.

2.2.1 Cost-Aware Platform

We assume the platform is cost-aware and is interested in collecting data with target quality levels (i.e. target privacy budget $\boldsymbol{\epsilon}$ for individuals) with minimum payment. The platform therefore uses an incentive mechanism R (also called a payment mechanism) such that $R_i(\mathbf{X})$ is the payment to individual i when the reported data is $\mathbf{X}$, which is a vector such that the ith entry is the reported data of individual i, i.e., X_i. The goal of the platform is to minimize the total payment $\sum_i R_i(\mathbf{X})$ under the constraint $\epsilon_i \geq \epsilon_i^\dagger$, where $\epsilon_i^\dagger$ is the target privacy budget chosen by the platform and ϵ_i is the actual quality level of the data from individual i. In other words, the platform aims at solving the following problem:

$$\min_R \mathbb{E}\left[\sum_i R_i(\mathbf{X})\right] \tag{2.3}$$

$$\text{subject to:} \quad \boldsymbol{\epsilon} \geq \boldsymbol{\epsilon}^\dagger \tag{2.4}$$

$$R_i(\mathbf{X}) \geq 0, \quad \forall \mathbf{X},\ \forall i. \tag{2.5}$$

We impose the constraint $R_i(\mathbf{X}) \geq 0$ for all $\mathbf{X}$ and all i so that negative payment (i.e., penalty) is not allowed, which is common in practice.

2.2.2 Strategic Individuals

We assume individuals are rational and strategic. Each individual is associated with a cost function $g_i(\epsilon)$ which is the cost incurred to individual i when the privacy budget is ϵ. We assume $g_i(\cdot)$ is an increasing function.

We assume individual i has the following information:

- cost function $g_i(\cdot)$,
- belief on θ, denoted by $\tilde{\theta}_i$ (we assume each individual has personalized $\tilde{\theta}_i$ to model her bias (or lack of information)),

- the payment mechanism announced by the platform, and
- the privacy preserving mechanism.

Let $R_i(X_i, \mathbf{X}_{-i})$ denote the payment received by individual i given reported data $\mathbf{X}$, which is simply a different notation for $R_i(\mathbf{X})$. The payment individual i expects to receive with privacy budget ϵ, based on her belief $\tilde{\theta}_i$, is

$$\begin{aligned} h_i(\epsilon) &= \mathbb{E}_{\tilde{\theta}_i}\left[R_i(X_i, \mathbf{X}_{-i})\right] \\ &= \sum_{\mathbf{x}\in\mathcal{X}^N} R_i(\mathbf{x}) \prod_j \left(\sum_{s\in\mathcal{S}} \mathbb{P}_{X|S}\left(x_j|s;\epsilon_j\right)\tilde{\theta}_{is} \right), \end{aligned}$$

where $\tilde{\theta}_{is}$ is individual i's belief on the probability that the private signal of individual j is s, and

$$\sum_{s\in\mathcal{S}} \mathbb{P}_{X|S}\left(x_j|s;\epsilon_j\right)\tilde{\theta}_{is}$$

is individual i's belief on the probability individual j reports x_j. Recall that we assume the private signals are independent across users.

Example: Consider the k-ary randomized response mechanism proposed in Kairouz *et al.* (2016) for discrete distribution estimation which guarantees differential privacy budget

$$\epsilon^{(d)} = \log(\epsilon + 1)$$

and is proved to be optimal in the low-privacy regime Kairouz *et al.* (2016). Under the k-ary randomized response mechanism, given $S = s$ and privacy budget ϵ,

$$\mathbb{P}_{X|S}(x|s;\epsilon) = \begin{cases} \frac{\epsilon+1}{\epsilon+M}, & x = s \\ \frac{1}{\epsilon+M}, & x \neq s \end{cases},$$

where $M = |\mathcal{S}|$. It is easy to see that the function is differentiable in ϵ. We can further obtain that

$$\sum_{s\in\mathcal{S}} \mathbb{P}_{X|S}\left(x_j|s;\epsilon_j\right)\tilde{\theta}_{is} = \frac{\epsilon_j\tilde{\theta}_{ix_j} + 1}{\epsilon_j + M}.$$

□

We assume individuals are strategic and are interested in maximizing the expected payoff, i.e. finding a quality level ϵ_i^* such that

$$\epsilon_i^* \in \arg\max_{\epsilon} \left(h_i(\epsilon) - g_i(\epsilon)\right). \tag{2.6}$$

Individuals are also rational so that they will not participate if

$$\max_{\epsilon} \left(h_i(\epsilon) - g_i(\epsilon)\right) < 0.$$

2.2.3 Minimum Payment Incentive Mechanism

Summarizing the discussions in the previous two subsections, the design of a minimum cost incentive mechanism is to solve the following problem:

$$\min_{R} \mathbb{E}\left[\sum_i R_i(\mathbf{X})\right] \tag{2.7}$$

$$\text{subject to: } R_i(\mathbf{X}) \geq 0, \quad \forall \mathbf{X},\ \forall i \tag{2.8}$$

$$\arg\max_{\epsilon} \left(h_i(\epsilon) - g_i(\epsilon)\right) \geq \epsilon_i^\dagger \quad \forall \tilde{\theta}_i, \forall i. \tag{2.9}$$

We next comment on constraint (2.9), which is called **B**ias-Proof condition in this chapter.

- **B**ias-Proof: We require condition (2.9) holds for all $\tilde{\theta}_i$ because the bias $\tilde{\theta}_i$ in general is unknown (or just partial known) to the platform. This condition guarantees that individual i chooses quality level at least $\epsilon_i^\dagger$ regardless of her bias, which we feel is important in practice where individuals often have only limited and heterogeneous knowledge about the underlying parameter θ.

2.3 A Winners-Take-All Incentive Mechanism

Before we present WINTALL, we first derive a lower bound on the payment to individual i with quality level ϵ. We define

$$\mathbb{P}_X(x;(\theta,\epsilon)) = \sum_{s\in\mathcal{S}} \mathbb{P}_{X|S}(x|s;\epsilon)\theta_s,$$

which is the probability that individual i reports x when the underlying parameter is θ and the privacy budget of individual i is ϵ.

To establish a lower bound, we relax the bias-proof constraint and assume θ is known to all data subjects:

$$\min_R \mathbb{E}\left[\sum_i R_i(\mathbf{X})\right] \tag{2.10}$$

$$\text{subject to: } R_i(\mathbf{X}) \geq 0, \quad \forall \mathbf{X},\ \forall i \tag{2.11}$$

$$\arg\max_{\epsilon} \left(h_i(\epsilon) - g_i(\epsilon)\right) \geq \epsilon_i^\dagger \quad \tilde{\theta}_i = \theta. \tag{2.12}$$

Let $L(\mathbf{X})$ denote the optimal solution to the problem.

Theorem 1. *Consider the optimization problem defined by* (2.10)-(2.12). *We have*

$$V_i^l\left(\epsilon_i^\dagger, \theta\right) \triangleq \min_{\epsilon\geq\epsilon_i^\dagger} g_i'(\epsilon) A(\epsilon,\theta) \leq \mathbb{E}\left[R_i(\boldsymbol{X})\right], \tag{2.13}$$

where

$$A(\epsilon,\theta) = \max_{x\in\mathcal{X}} \left\{ \frac{\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon))}{\partial\epsilon}}{\mathbb{P}_X(x;(\theta,\epsilon))} \right\}. \tag{2.14}$$

□

The proof of this theorem is in the appendix. We remark that since the optimization problem (2.10)-(2.12) is a relaxed version of problem (2.7)-(2.9), $V_i^l(\epsilon,\theta)$ is a lower bound on the amount of payment to data subject i for reporting data with privacy budget at least ϵ.

Given the lower bound, the question now is whether the lower bound can be achieved? To answer this question, we first introduce the following notations:

$$x^{\diamond}(\theta,\epsilon) \in \arg\max_{x\in\mathcal{X}} \left\{ \frac{\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon))}{\partial \epsilon}}{\mathbb{P}_X(x;(\theta,\epsilon))} \right\}, \tag{2.15}$$

and

$$x^*(\theta,\epsilon_i^\dagger) = x^{\diamond}(\theta,\epsilon_i^*), \tag{2.16}$$

where

$$\epsilon_i^* = \arg\min_{\epsilon\geq\epsilon_i^\dagger} g_i'(\epsilon)A(\epsilon,\theta). \tag{2.17}$$

Furthermore, define

$$W_i(\theta,\epsilon_i^\dagger) \triangleq \frac{g_i'(\epsilon_i^*)}{\frac{\partial \mathbb{P}_X(x^{\diamond}(\theta,\epsilon_i^*);(\theta,\epsilon_i^*))}{\partial \epsilon}}. \tag{2.18}$$

Note that the lower bound can be written as

$$\min_{\epsilon\geq\epsilon_i^\dagger} \frac{g_i'(\epsilon)}{\frac{\partial \mathbb{P}_X(x^{\diamond}(\theta,\epsilon);(\theta,\epsilon))}{\partial \epsilon}} \mathbb{P}_X(x^{\diamond}(\theta,\epsilon);(\theta,\epsilon)). \tag{2.19}$$

Suppose ϵ is the optimal solution to the problem above. Then it suggests that the lower bound can be achieved by paying individual i only when she reports $x_{\theta,\epsilon}^{\diamond}$ with a payment of

$$\frac{g_i'(\epsilon)}{\frac{\partial \mathbb{P}_X(x^{\dagger}(\theta,\epsilon);(\theta,\epsilon))}{\partial \epsilon}}. \tag{2.20}$$

We remark while (2.19) involves an optimization problem with respect to ϵ, it is the average payment a data subject expects to receive given privacy budget ϵ. A desired property of a payment mechanism (together the privacy protection mechanism) is to have the average payment be an increasing function of ϵ. In other words, a more accurate reporting should result in higher expected return. If it is the case, then the optimal solution to (2.19) is $\epsilon_i^\dagger$, i.e. the target privacy level. Furthermore, when this property holds and the target privacy budget is the same for all individuals, then $x_{\theta,\epsilon}^*$ is the same for all individuals as well. The payment could differ when privacy

cost functions are different. In the following section, we will present conditions and examples that the desired property mentioned above holds.

The lower bound above suggests the following incentive mechanism.

A Winners-Take-All Incentive Mechanism (WINTALL)

(1) The platform announces target quality level $\boldsymbol{\epsilon}^{\dagger}$.

(2) Each individual reports her data (which can also be an decision of not participating).

(3) For non-participating individual, the payment is zero.

(4) From reported data $\mathbf{X}$, the platform estimates θ, denoted by $\tilde{\theta}$.

(5) For each participating individual i, the platform pays according to the reported X_i, the estimation $\tilde{\theta}$, and the target quality level $\epsilon_i^{\dagger}$. Specifically, if the reported data is $x^*\left(\tilde{\theta}, \epsilon_i^{\dagger}\right)$, individual i receives a payment of $W_i\left(\tilde{\theta}, \epsilon_i^{\dagger}\right)$; otherwise, no payment is made to individual i. □

Now let us understand whether the proposed incentive mechanism actually achieves the lower bound asymptotically. Note that instead of tagging each value a fixed price, the payment under WINTALL is based on the reported data and the estimation. So it is a joint learning and incentive mechanism. Since the payment depends on the estimation, the efficiency of WINTALL depends on the number of data subjects in the system. When there are a large number of data subjects, the data collector can accurately estimate θ, so the payment made to each data subject is close to the lower bound.

We first prove the following theorem, which shows that $\boldsymbol{\epsilon}^{(t)}$ is a bias-proof Nash equilibrium under WINTALL. Note that under WINTALL, given the estimation $\tilde{\theta}$, the payment to individual i is independent of other individuals' reports. Furthermore,

individual i needs to decide on the quality level ϵ_i before they receive their private data. Therefore, we assume that individual i is confident about her belief and uses her belief on the distribution $\tilde{\theta}_i$ when choosing the quality level ϵ_i.

Theorem 2. *If for any i,*

$$W_i\left(\theta, \epsilon_i^{(t)}\right) \mathbb{P}_X\left(x^*\left(\theta, \epsilon\right); (\theta, \epsilon)\right) - g_i(\epsilon)$$

is strictly concave in ϵ for given $\boldsymbol{\epsilon}^{(t)}$ and any θ, then the target quality level $\boldsymbol{\epsilon}^{(t)}$ is a bias-proof Nash equilibrium under WINTALL. With quality level $\epsilon_i^{(t)}$, the expected payment individual i receives is

$$W_i(\theta, \epsilon)\mathbb{P}_X\left(x^*\left(\theta, \epsilon\right); (\theta, \epsilon)\right), \tag{2.21}$$

which equals to $V_i^l\left(\epsilon_i^{(t)}, \theta\right)$ when $\tilde{\theta} = \theta$ (i.e. the platform can accurately estimate Θ from the collected data). □

The proof of this theorem is in the appendix.

‘‘Winners-Take-All”: Suppose $\{S_i\}$ are identically distributed and the target quality level is the same for all individuals. In this case, $x^*_{i,\tilde{\theta},\epsilon_i^{(t)}}$ is independent of i and can be written as $x^*_{\tilde{\theta},\epsilon}$. Therefore, only individuals who report $x^*_{\tilde{\theta},\epsilon}$ will be paid. In other words, under WINTALL, after collecting all data, the platform determines a “winning” report $x^*_{\tilde{\theta},\epsilon}$ and all payments go to the “winners”.

We now consider WINTALL under the k-ary randomized response mechanism proposed in Kairouz *et al.* (2016) for discrete distribution estimation which guarantees differential privacy budget

$$\epsilon^{(d)} = \log(\epsilon + 1) :$$

$$\mathbb{P}_{X|S}(k|m;\epsilon) = \begin{cases} \frac{\epsilon+1}{\epsilon+M}, & k = m \\ \frac{1}{\epsilon+M}, & k \neq m \end{cases}.$$

Given the k-ary randomized response mechanism, we have

$$\begin{aligned}\mathbb{P}_X(m;(\epsilon;\theta)) &= \sum_{k=1}^{M} \mathbb{P}_{X|S}(m|k;(\epsilon,\theta))\mathbb{P}_S(k;(\epsilon,\theta)) \\ &= \theta_m \frac{\epsilon+1}{\epsilon+M} + (1-\theta_m)\frac{1}{\epsilon+M} \\ &= \frac{\theta_m \epsilon + 1}{\epsilon+M},\end{aligned}$$

and

$$\frac{\partial}{\partial\epsilon}\mathbb{P}_X(m;(\epsilon;\theta)) = -\frac{\theta_m\epsilon+1}{(\epsilon+M)^2} + \frac{\theta_m}{\epsilon+M} = \frac{\theta_m M - 1}{(\epsilon+M)^2}.$$

Therefore, we have

$$\begin{aligned}\frac{\mathbb{P}_X(m;(\epsilon;\theta))}{\frac{\partial}{\partial\epsilon}\mathbb{P}_X(m;(\epsilon;\theta))} &= \frac{\epsilon\theta_m+1}{\epsilon+M}\frac{(\epsilon+M)^2}{M\theta_m-1} \\ &= (\epsilon+M)\frac{\epsilon}{M}\left(1+\frac{\frac{1}{\epsilon}+\frac{1}{M}}{\theta_m-\frac{1}{M}}\right),\end{aligned}$$

which is a decreasing function of θ_m. From that, we conclude that

$$m^* = \arg\min_m \frac{\mathbb{P}_X(m;(\epsilon;\theta))}{\frac{\partial}{\partial\epsilon}\mathbb{P}_X(m;(\epsilon;\theta))} = \arg\max_m \theta_m.$$

Note that unless $\theta_m = \frac{1}{M}$ for all m, i.e., a uniform distribution, we have $\theta_{m^*} > \frac{1}{M}$, which implies that

$$\frac{\partial}{\partial\epsilon}\mathbb{P}_X(m^*;(\epsilon;\theta)) = \frac{\theta_{m^*}M-1}{(\epsilon+M)^2} > 0,$$

and $\mathbb{P}_X(m^*;(\epsilon;\theta))$ is strictly concave in ϵ because $\frac{\partial}{\partial\epsilon}\mathbb{P}_X(m^*;(\epsilon;\theta))$ is a decreasing function in ϵ. We also note that for any $\epsilon > 0$,

$$m^* = \arg\max_m \theta_m = \arg\max_m \mathbb{P}_X(m;(\theta,\epsilon)).$$

In other words, the most popular answers in the private data and in the reported data are the same.

WINTALL with a target quality level ϵ in this case is as follows.

WINTALL for Private Discrete Distribution Estimation

- After collecting data from N individuals, denoted by $\{x_i\}_{i=1,\cdots,N}$, the platform identifies the most popular answer m^* :

$$m^* \in \arg\max_m \frac{\sum_{i=1}^N 1_{x_i=m}}{N}.$$

Ties are broken uniformly at random.

- The platform pays each user who reports m^* an amount of

$$\frac{\partial g(\epsilon)}{\partial \epsilon} \frac{(M+\epsilon)\epsilon}{M\frac{\sum_{i=1}^N 1_{x_i=m^*}}{N} - 1}. \tag{2.22}$$

□

Remark: We note that the most popular answer in the private data is consistent with that in the reported data, which creates the incentive for an individual to report an answer close to her private data because the individual expects her private answer to be the dominating one. Note that in this example, the only prior information the platform needs is $\frac{\partial g(\epsilon)}{\partial \epsilon}$.

2.4 Proof of Theorem 1

Recall that the relaxed optimization problem (2.10)-(2.11) assumes each data subject knows θ. Under a nonnegative payment mechanism R, the expected payment to individual i is

$$\begin{aligned}
&\mathbb{E}[R_i(X_i, \mathbf{X}_{-i})] \\
&= \sum_{\mathbf{x}\in\mathcal{X}^N} R_i(x_i, \mathbf{x}_{-i}) \mathbb{P}_{\mathbf{X}}(\mathbf{x}; (\theta, \boldsymbol{\epsilon})) \\
&= \sum_{x_i\in\mathcal{X}} \mathbb{P}_{X_i}(x_i; (\theta, \epsilon_i)) \sum_{\mathbf{x}_{-i}\in\mathcal{X}^{N-1}} R_i(x_i, \mathbf{x}_{-i}) \mathbb{P}_{\mathbf{X}_{-i}}(\mathbf{x}_{-i}; (\theta, \boldsymbol{\epsilon}_{-i})).
\end{aligned}$$

Define

$$\bar{R}_i(x_i, (\boldsymbol{\epsilon}_{-i}, \theta)) = \sum_{\mathbf{x}_{-i}\in\mathcal{X}^{N-1}} R_i(x_i, \mathbf{x}_{-i}) \mathbb{P}_{\mathbf{X}_{-i}}(\mathbf{x}_{-i}; (\theta, \boldsymbol{\epsilon}_{-i})),$$

which is the expected payment data subject i receives when reporting x_i. Then we have

$$\mathbb{E}[R_i(X_i, \mathbf{X}_{-i})] = \sum_{x_i \in \mathcal{X}} \mathbb{P}_X(x_i; (\theta, \epsilon_i)) \bar{R}_i(x_i, (\boldsymbol{\epsilon}_{-i}, \theta)).$$

Let $\hat{R}_i(x_i, \theta)$ denote a genie-aided payment mechanism which knows the parameter θ, and pays individual i based on θ and x_i. If the genie-aided mechanism pays individual i an amount of $\bar{R}_i(x_i, \epsilon_{-i}, \theta)$ when individual i reports x_i, then $\hat{R}_i(x_i, \theta)$ and $R_i(\mathbf{X})$ have the same expected payment to individual i. Therefore, any payment mechanism based on the reported data $\mathbf{x}$ can be mimicked by a genie-aided payment mechanism with the same expected payment.

Now if the optimal privacy budget of individual i is ϵ_i ($\epsilon_i \geq \epsilon_i^\dagger$) then the following equation has to hold

$$\frac{\partial}{\partial \epsilon} \mathbb{E}[R_i(X_i, \mathbf{X}_{-i})] - g_i'(\epsilon_i) \tag{2.23}$$

$$= \sum_{x_i \in \mathcal{X}} \frac{\partial \mathbb{P}_X(x_i; (\theta, \epsilon_i))}{\partial \epsilon} \bar{R}_i(x_i, (\boldsymbol{\epsilon}_{-i}, \theta)) - g_i'(\epsilon_i) \tag{2.24}$$

$$= 0. \tag{2.25}$$

Since $R(\mathbf{X})$ can be mimicked by a genie-aided payment mechanism and (2.25) is a necessary condition for ϵ_i to be the optimal privacy budget, the solution to problem (2.7) is lower bounded by the solution to the following minimum payment problem:

$$\min_{\hat{R}} \sum_{x \in \mathcal{X}} \hat{R}_i(x, \theta) \mathbb{P}_X(x; (\theta, \epsilon_i))$$
$$\text{subject to:} \sum_{x \in \mathcal{X}} \hat{R}_i(x, \theta) \frac{\partial \mathbb{P}_X(x; (\theta, \epsilon_i))}{\partial \epsilon} - g_i'(\epsilon_i) = 0.$$

Note that

$$\sum_{x \in \mathcal{X}} \hat{R}_i(x, \theta) \mathbb{P}_X(x; (\theta, \epsilon_i))$$
$$= \sum_{x \in \mathcal{X}} \hat{R}_i(x, \theta) \frac{\partial \mathbb{P}_X(x; (\theta, \epsilon_i))}{\partial \epsilon} \frac{\mathbb{P}_X(x; (\theta, \epsilon_i))}{\frac{\partial \mathbb{P}_X(x; (\theta, \epsilon_i))}{\partial \epsilon}}.$$

Now recall the definition of $A(\epsilon, \theta)$, we have

$$\begin{aligned}
&\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}>0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\frac{\mathbb{P}_X(x;(\theta,\epsilon_i))}{\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}} \\
\geq &\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}>0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}A(\epsilon_i,\theta). \qquad (2.26)
\end{aligned}$$

Note that $A(\epsilon_i, \theta) > 0$ and $\hat{R}_i(x, \theta) \geq 0$ according to their definitions, so

$$\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\leq 0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}A(\epsilon_i,\theta) \leq 0.$$

On the other hand,

$$\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\leq 0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\frac{\mathbb{P}_X(x;(\theta,\epsilon_i))}{\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}} \geq 0,$$

which implies

$$\begin{aligned}
&\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\leq 0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\frac{\mathbb{P}_X(x;(\theta,\epsilon_i))}{\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}} \\
\geq &\sum_{x\in\mathcal{X}:\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}\leq 0} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}A(\epsilon_i,\theta). \qquad (2.27)
\end{aligned}$$

Combining inequalities (2.26) and (2.27), we have

$$\begin{aligned}
&\sum_{x\in\mathcal{X}} \hat{R}_i(x,\theta)\mathbb{P}_X(x;(\theta,\epsilon_i)) \\
\geq &\sum_{x\in\mathcal{X}} \hat{R}_i(x,\theta)\frac{\partial \mathbb{P}_X(x;(\theta,\epsilon_i))}{\partial\epsilon}A(\epsilon_i,\theta) \\
= &g_i'(\epsilon_i)A(\epsilon_i,\theta),
\end{aligned}$$

where the last equality holds due to condition (2.25), which immediately leads to the lower bound.

2.5 Proof of Theorem 2

Suppose individual i believes that $\tilde{\Theta}_i$ is the estimation of the platform, the payment of individual i expects is

$$\mathbb{E}[R_i(X_i, \mathbf{X}_{-i})|\epsilon_i] = \int_{\hat{\theta}} W_{i,\epsilon_i^{(t)},\hat{\theta}} \mathbb{P}_{X_i} \left(x^*_{i,\epsilon_i^{(t)},\hat{\theta}}; (\hat{\theta}, \epsilon_i) \right) f_{\tilde{\Theta}_i}(\hat{\theta})\, d\hat{\theta},$$

and the expected payoff is

$$\begin{aligned} &\mathbb{E}[R_i(X_i, \mathbf{X}_{-i})|\epsilon_i] - g_i(\epsilon_i) \\ =& \int_{\hat{\theta}} \left(W_{i,\epsilon_i^{(t)},\hat{\theta}} \mathbb{P}_{X_i} \left(x^*_{i,\epsilon_i^{(t)},\hat{\theta}}; (\hat{\theta}, \epsilon_i) \right) - g_i(\epsilon_i) \right) f_{\tilde{\Theta}_i}(\hat{\theta})\, d\hat{\theta}. \end{aligned}$$

Note that under the assumption of the theorem,

$$W_{i,\epsilon_i^{(t)},\hat{\theta}} \mathbb{P}_{X_i} \left(x^*_{i,\epsilon_i^{(t)},\hat{\theta}}; (\hat{\theta}, \epsilon_i) \right) - g_i(\epsilon_i)$$

is strictly concave in ϵ_i, and furthermore

$$W_{i,\epsilon_i^{(t)},\hat{\theta}} \mathbb{P}_{X_i} \left(x^*_{i,\epsilon_i^{(t)},\hat{\theta}}; (\hat{\theta}, \epsilon_i) \right) - g_i(\epsilon_i) \Big|_{\epsilon_i = \epsilon_i^{(t)}} = 0$$

according to the definition of $W_{i,\epsilon_i^{(t)},\hat{\theta}}$. Therefore,

$$\epsilon_i^{(t)} = \arg\max_{\epsilon_i} W_{i,\epsilon_i^{(t)},\hat{\theta}} \mathbb{P}_{X_i} \left(x^*_{i,\epsilon_i^{(t)},\hat{\theta}}; (\hat{\theta}, \epsilon_i) \right) - g_i(\epsilon_i)$$

which holds for any $\hat{\theta}$. Hence,

$$\epsilon_i^{(t)} = \arg\max_{\epsilon_i} \mathbb{E}[R_i(X_i, \mathbf{X}_{-i})|\epsilon_i] - g_i(\epsilon_i),$$

which implies $\boldsymbol{\epsilon}^{(t)}$ is a Nash equilibrium. It is easy to check that (2.21) matches the lower bound when $\tilde{\theta} = \theta$.

2.6 Experiments on Amazon Mechanical Turk

In this section, we present experimental results using Amazon Mechanical Turk (MTurk). Because of the logistical and legal matters involved in collecting sensitive

personal data, we conducted an alternative experiment to evaluate the effectiveness of WINTALL as well. From the perspective of users, this chapter studies the trade off between privacy and payment. A similar trade off exists in crowd-sourcing where the quality of a task increases as a worker invests more time and effort to finish the task. So we can draw a close equivalence between these problems as shown in Table 2.1. Our results show that WINTALL-type incentive mechanisms can help platform obtain better answers with less payment compared with the traditional flat-rate-payment mechanism that pay a flat-fee to each worker who participates.

Private-data Collection	Crowd-Sourcing
Privacy-level in reported data	Job quality
Privacy-Loss	Time and Effort

Table 2.1: The Equivalence of Private-Data Collection and Crowd-Sourcing

Our tasks were to translate paragraphs written in English to Chinese. This corresponds the scenario that the ground true has a large alphabet set and θ is also unknown. We acknowledge that to calculate the "optimally" payment under WINTALL, the data collector needs to know θ and the cost functions of data subjects, both of which may not be available. Our experiment was designed to show the significant gain of WINTALL even without knowing these pieces of information.

The key intuition behind WINTALL is only to pay users who provide the "best" answer, where the quality of an answer is evaluated by comparing with answers submitted by other users. In this experiment, we will therefore compare all translations submitted by workers and then offer significant bonus to user(s) who provided the best translation. In particular, we launched 9 different translation tasks on MTurk. Each task is to translate a paragraph of 90-120 words chosen from some articles

about IT from Economist.com to Chinese. For each task, we tested the following three different payment mechanisms, where payment amounts were chosen according to typical payments at MTurk, which is 10 to 100 cents.

- Flat: The first payment mechanism pays each worker 30 cents after finishing a task.
- WINTALL-20: The second payment mechanism pays each worker 20 cents after finishing each task and additional 20 cents if the translation is considered to be good.
- WINTALL-40: The third mechanism pays each worker 20 cents after finishing each task and additional 40 cents if the translation is considered to be good.

Note that each payment mechanism includes a base payment so that workers would not view the tasks as scams. We can think each task offer 20 cents as participation fee so the first mechanism pays 10 cents for each completed task, the second pays 20 cents only to high-quality translations and the third one pays 40 cents only to high-quality translations. Here are two key observations:

- Comparing Flat and WINTALL-20, WINTALL-20 results in more high-quality translations with *less* total payment.
- Comparing WINTALL-20 and WINTALL-40, WINTALL-40 results in more high-quality translations.

Both observations are consistent with our theoretical results. More details about the experiment are presented below.

Experiment Setup

For each translation task and each payment mechanism, we set up 20 hits, which means at most 20 workers are invited to finish each translation task under a specific

payment mechanism, we have $3 \times 9 \times 20 = 540$ data samples. Here is one of the translation task:

"*Will that be enough? European privacy activists are derisive about the new arrangement. They do not believe that an American government agency which operates in secret can be trusted to obey any rules. What is the point in Europeans having judicial redress when they will not know if their data has been spied on? It is likely that the new deal will be tested in the European Court of Justice. But if the European Commission tells the court that American privacy protection is now adequate, it will be a lot harder for judges to rule otherwise. The transatlantic data flows, and those whose jobs, profits and deals depend on them, look a lot safer.*"

After the experimental results have been collected, they were evaluated by the first author who is a native Chinese. We rejected some answers not written in Chinese or with extremely poor quality (these are only a few of them).

To evaluate the quality of each translation, we graded each translation from 0 to 5. Each paragraph consists of 4 to 6 sentences. We gave one point if one sentence is correctly translated. Sometimes, one point was given to half of a sentence if the paragraph consists of less than 5 sentences, or one point was given to two sentences if the paragraph consists of more than 5 sentences.

Figures 2.1 and 2.2 show the average scores and average payments of the nine different tasks under different payment mechanisms. As we can see from Figure 2.1, for 7 of 9 tasks, WINTALL-20 yielded higher average scores than Flat, 25.4% increase on average, and for all 9 tasks, WINTALL-40 resulted in higher mean scores than Flat, 47.3% increase on average. Comparing WINTALL-20 and WINTALL-40, for 7 of the 9 tasks, WINTALL-40 resulted in higher mean scores, 17.7% increase on average. Figure 2.2 shows that for all tasks both WINTALL-20 and WINTALL-40 resulted in lower average payments than Flat. When comparing WINTALL-20 and

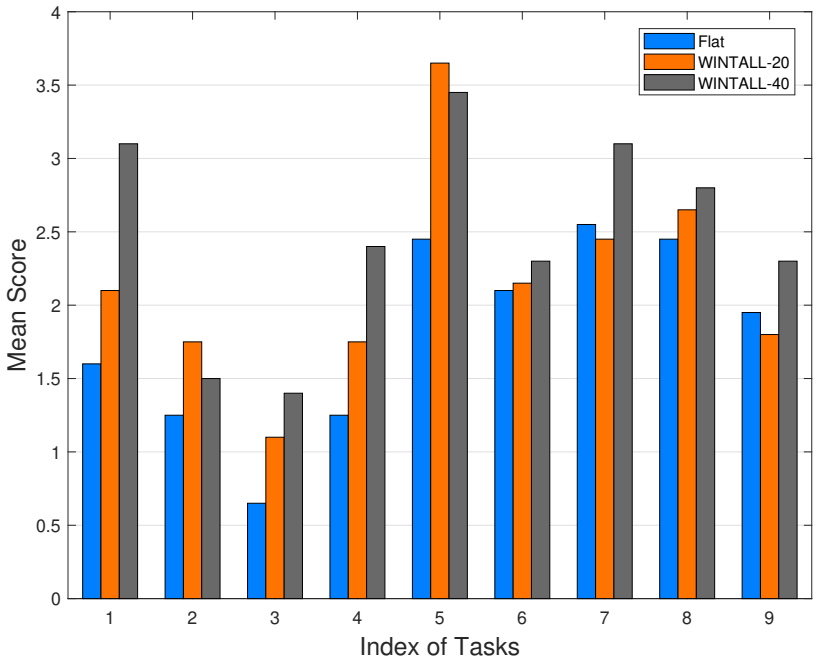

Figure 2.1: Mean Scores of Different Tasks with Different Payment Mechanisms

WINTALL-40, WINTALL-40 paid more in 8 of 9 tasks.

Figure 2.3 shows the distribution of scores across the 9 tasks under different payment mechanisms. We can clearly see that the distribution shifts towards higher scores when the bonus payment increases. The number of good answers (score 4 or 5) increases from 16 for Flat to 28 for WINTALL-20 and 37 for WINTALL-40, respectively. The number of very good answers (score 5) increases from 7 (Flat) to 9 (WINTALL-20) and 12 (WINTALL-40), respectively. Figure 2.4 shows that average scores across all tasks under different payment mechanisms. Comparing to Flat, the average score increases by 19.4% and 37.5%, respectively, while the average payment reduces by 30% and 24.4%, respectively. The results of this experiment confirmed the novelty of our proposed payment mechanism.

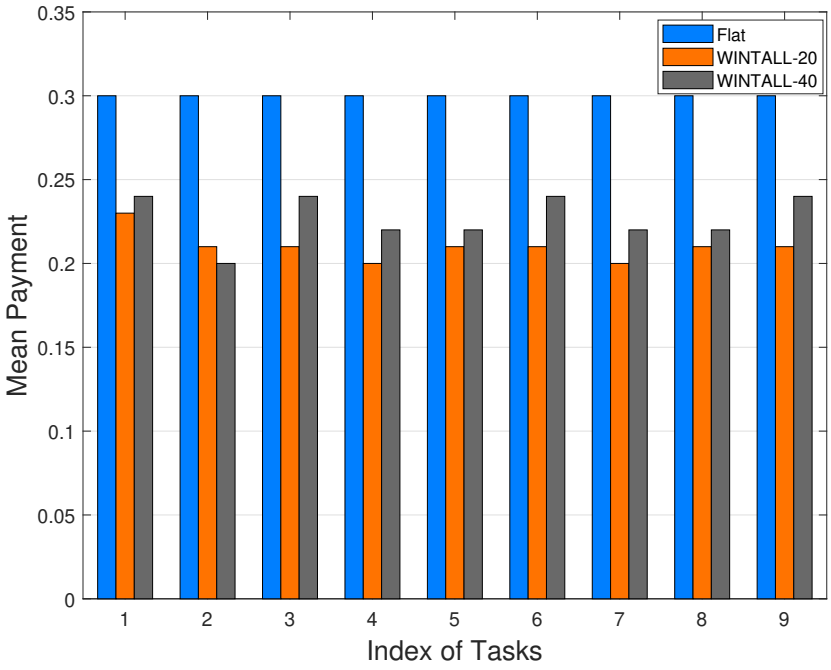

Figure 2.2: Mean Payment of Different Tasks with Different Payment Mechanisms

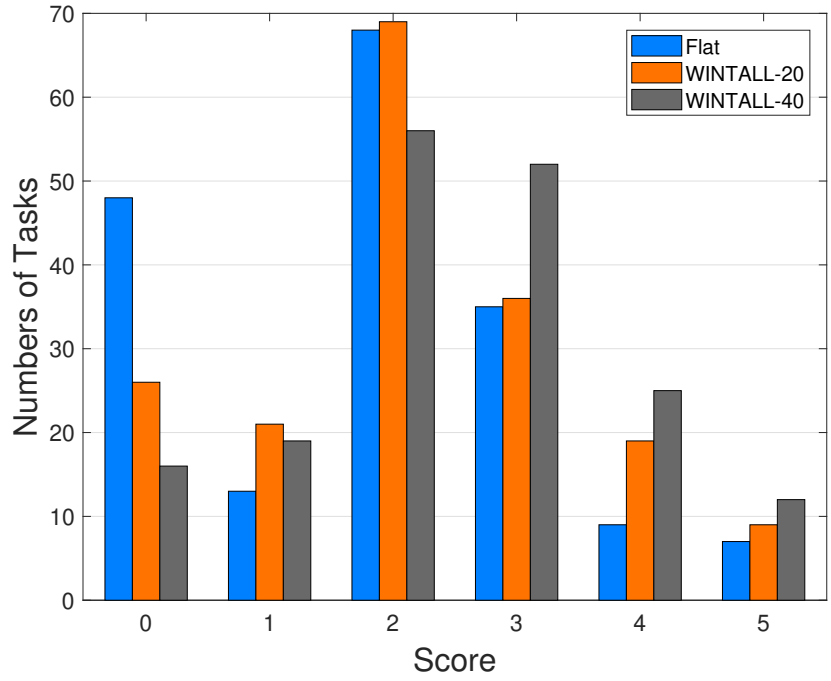

Figure 2.3: Score Distributions under Different Payment Mechanisms

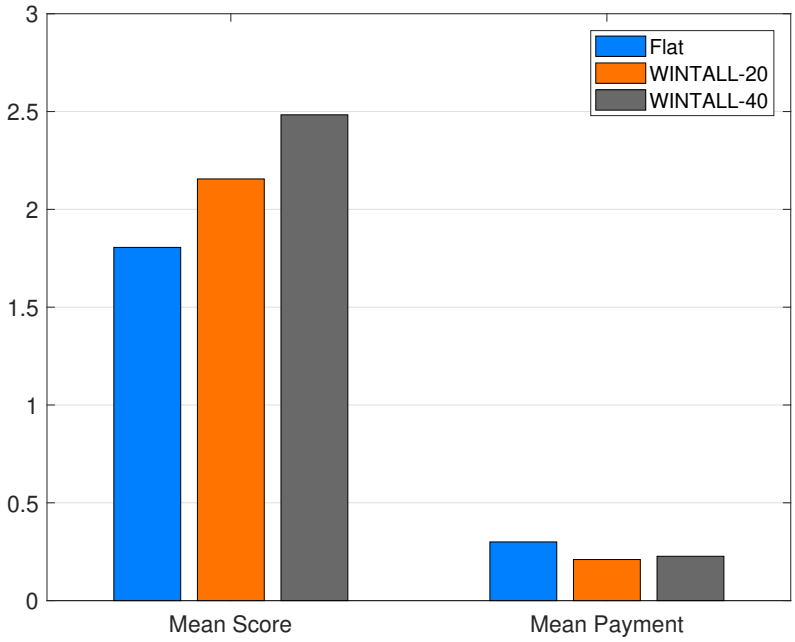

Figure 2.4: Mean Score and Mean Payment under Different Payment Mechanisms

2.7 Summary

In this chapter, we studied incentive mechanisms for private discrete distribution estimation. We first derived a lower bound on the minimum payment required for guaranteeing quality level, and then proposed WINTALL — a novel incentive mechanism. The expected payment of WINTALL matches the lower bound when the underlying parameter θ can be estimated by the platform accurately. We present its application to private discrete distribution estimation, where WINTALL rewards individuals whose reported answers match the most popular one. We also presented real-world experiments on Amazon Mechanical Turk to validate the novelty of WINTALL-inspired mechanisms.

Chapter 3

AN OPTIMAL STOPPING APPROACH FOR ITERATIVE TRAINING IN FEDERATED LEARNING

The chapter considers an important sub-problem in federated learning: when should the parameter server stop the current round (iteration), updates the machine learning model and starts the next round.

Federated learning Training process is run on distributed device so that a device does not need to expose personal data on the device to servers or other devices. The updates for the model (e.g. the gradients of SGD) will be transmitted to a parameter server which will aggregate the updates to update the machine learning model. The updated model will then be broadcast to the devices for the next iteration of training. So when the parameter server should stop the current iteration, updates the machine learning model and starts the next iteration becomes an important problem to be figured out.

Such a decision is based on the number of updates received, the expected waiting time to receive the next update,and how the loss function decreases as the number of updates increases.

3.1 Optimal stopping problem formulation in federated learning

In this section, we introduce how we formulate the problem as an optimal stopping problem and propose an optimal solution for this problem .

We consider a system with a single parameter server and M client devices such as mobile phones, where each client owns a local dataset. The system is used to train a learning model using the local datasets in an iterative fashion. Each iteration is

called a "round". At the beginning of each round, the parameter server broadcasts the latest parameters (such as the parameters of the neural network) to the clients. After receiving the parameters, each client trains the model using its local dataset, e.g. calculate the gradients using SGD, and then transmits the updates (e.g. the gradients) to the parameter server, which aggregates the updates to obtain a new model. This finishes one round, and the next round starts when the parameter server broadcasts the new parameters to the clients. We further assume following idealized data processing and communication models.

Data Processing Model: We assume each client finishes the data processing and computing the update with some probability p at each time slot. In other words, we assume the processing time of each dataset is geometrically distributed.

Communication Model: We assume the client mobile devices share a single channel when communicating with the parameter server (this assumption can be easily extended to multichannel OFDM systems). At the beginning of a time slot, one of the clients who have finished their computing tasks but have not transmitted the data to the parameter server will be selected uniformly at random to upload the update to the server, and the transmission succeeds with probability μ at the end of the time slot. At some stopping time (the choice of the stopping time is the focus of this paper), the parameter server stops accepting new updates and updates the global model using all uploaded information. The parameter server then broadcasts a new global model to all clients to start a new round.

This iterative training process is shown in Figure 3.1, where τ denotes length of a time slot.

We assume the amount of time it takes for the parameter server to broadcast the updated parameters is t_0, which remains a constant for all rounds and includes both the time it takes to aggregate all the updates it receives and the time it takes to

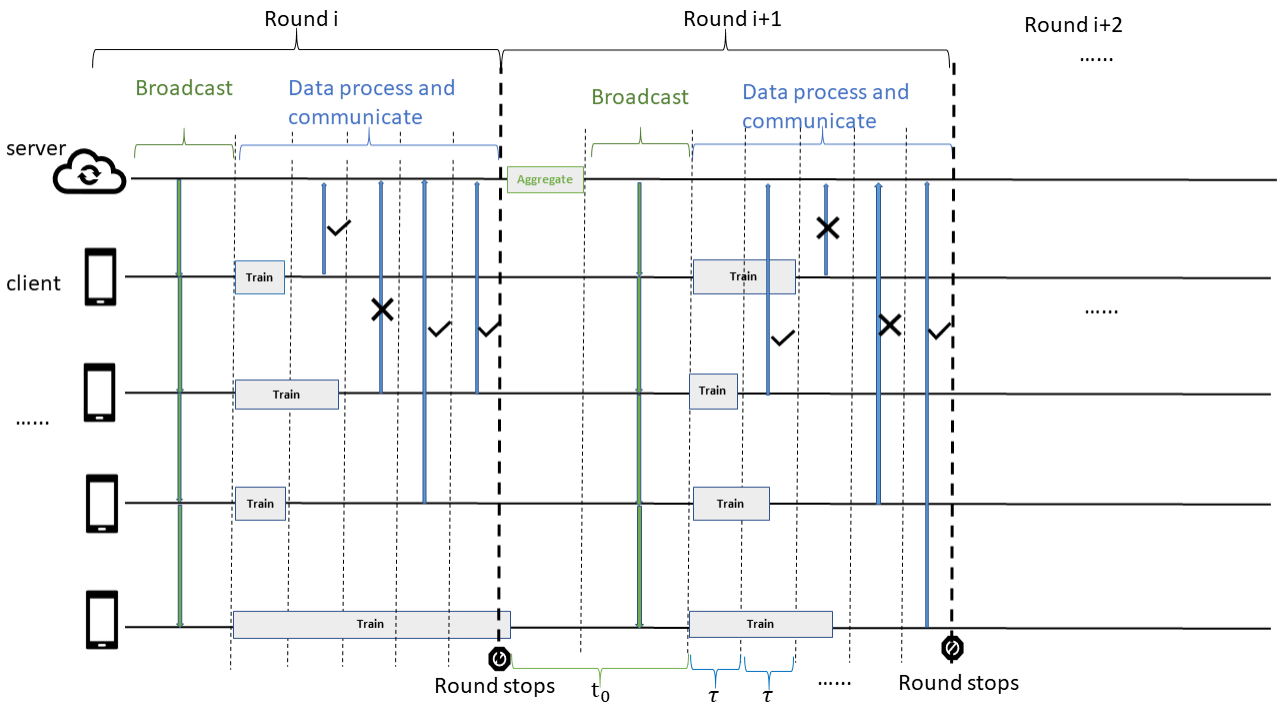

Figure 3.1: Iterative Training Process

transmit the new parameter to all clients.

We define α_n is the number of clients who have not finished processing their local datasets, β_n is the number of clients who have finished computing but have not transmitted the updates to the parameter server, and $k_n = M - \alpha_n - \beta_n$ is the number of clients who have updated the parameters based on local datasets and also uploaded the updates to the parameter server. The iterative training process can be modeled as a discrete-time Markov chain (DTMC) whose state at time slot n is denoted by $X_n = [K_n, \boldsymbol{\beta}_n]$. Denote by $x_n = [k_n, \beta_n]$ is a realization of X_n.

Let $R(k)$ denote the reward that the parameter server obtains after receiving the kth update. The reward R can be decrement of the loss function. We make the following assumptions in this paper:

(1) R is positive, increasing with k and bounded.

(2) $\Delta R(k) = R(k+1) - R(k)$ is decreasing in k, i.e. the reward of receiving a new update is diminishing as the parameter receives more and more updates.

We first focus on a single round with a given reward function. Let N denote the number of time slots in this ground, which is a random variable as the parameter server can decide to terminate this round and start the next round anytime. We consider the following stopping problem:

$$J^* = \sup_{\pi \in S} \frac{\mathbb{E}[R(K_N)]}{\mathbb{E}[N\tau + t_0]} \tag{3.1}$$

where π is a stopping policy, $\mathcal{S}$ is the set of all stopping policy, N is stopping time, t_0 is a constant as defined above.

The problem above is difficult to solve. Instead, we introduce the following problem and then show that resolving this new problem can lead to the solution of the original problem:

$$V_\lambda = \sup_{\pi \in \mathcal{S}} \mathbb{E}\left[R(K_N) - \lambda(N\tau + t_0)\right] \tag{3.2}$$

where λ is a positive constant.

For simplicity, we include the time slot in the system state $(X_n, n) = (K_n, \boldsymbol{\beta}_n, n)$. The transition probabilities of the Markov chain are as follows, for given state (k, β, n) and any $0 \leq i \leq M - k - \beta$:

If $\beta > 0$, then

$$\begin{aligned}
&\Pr\left[(X_{n+1}, n+1) = (k, \beta + i, n+1) | (X_n, n) = (k, \beta, n)\right] \\
=&(1-\mu)\binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i},
\end{aligned}$$

$$\begin{aligned}
&\Pr\left[(X_{n+1}, n+1) = (k+1, \beta - 1 + i, n+1) | (X_n, n) = (k, \beta, n)\right] \\
=&\mu\binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i};
\end{aligned}$$

If $\beta = 0$, then

$$\begin{aligned}&\Pr\left((X_{n+1}, n+1) = (k, i, n+1) | (X_n, n) = (k, 0, n)\right)\\ =&\binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i}.\end{aligned}$$

If $\beta > 0$, then:

$$\Pr\left[(k, \beta + i, n+1) | (k, \beta, n)\right] = (1-\mu)\binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i},$$

$$\Pr\left[(k+1, \beta - 1 + i, n+1) | (k, \beta, n)\right] = \mu\binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i};$$

If $\beta = 0$, then:

$$\Pr\left((k, i, n+1) | (k, 0, n)\right) = \binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i}.$$

We define $V(\cdot)$ to be the value function so that $V(k, \beta, n)$ is the value of state $(x_n, n) = (k, \beta, n)$, and

$$V(k, \beta, n) = \sup_{\pi, K_N \leq M-k} \mathbb{E}\left[R(k + K_N) - \lambda((N+n)\tau + t_0)\right].$$

We can easily verify that $V_\lambda = V(0, 0, 0)$.

The following theorem establishes the relationship between the original problem and the modified problem. The proof can be found in the appendix.

Theorem 3. *If there exists λ such that*
$V_\lambda = \sup_{\pi \in S} \mathbb{E}(R(K_N) - \lambda(N\tau + t_0)) = 0$*, then*

$$J^* = \sup_{\pi \in S} \frac{\mathbb{E}[R(K_N)]}{\mathbb{E}[\lambda(N\tau + t_0)]} = \lambda.$$

Furthermore, if $V_\lambda = \sup_{\pi \in S} \mathbb{E}(R(K_N) - \lambda(N\tau + t_0)) = 0$ is attained by some policy $\pi^* \in S$, then the policy π^* is also optimal for maximizing $\mathbb{E}\left[R(K_N)\right] / \mathbb{E}\left[N\tau + t_0\right]$.

Motivated by the result above, we can find the optimal parallel training policy by the following steps.

- First initialize λ, and find optimal π_λ for Problem (3.2) as well as V_λ.
- Repeatedly update λ, and find a new optimal policy π_λ and optimal value V_λ for Problem (3.2) until finding a λ^* such that optimal value $V_{\lambda^*} = 0$. The final policy π^* is the optimal policy.

In next section, we present a low complexity algorithm for Problem (3.2).

3.2 Low-Complexity Algorithm for Solving the Modified Optimal Stopping problem

In this section, we focus on fixed M and given any λ. We will show that optimal policy is a threshold policy. For any state $(x_n, n) = (k, \beta, n)$, the actions are to either terminate the current round or to continue to the next time slot. The reward of stopping at that stage is $R(k) - \lambda(n \cdot \tau + t_0)$. If $\beta = 0$, then to continue means to let unfinished clients continue to compute, but no update will be transmitted since $\beta = 0$. Otherwise, $\beta_n \geq 1$, to continue means a randomly selected client from the β clients will transmit its result to the server while $(M - k - \beta)$ unfinished clients continue to process their datasets. The Bellman equation in these two cases can be written as

If $\beta > 0$:

$$\begin{aligned} V(k,\beta,n) = \max\{&R(k) - \lambda(n\tau + t_0), \\ &\mu\left(\sum_{i=0}^{M-k-\beta} W(i)V(k+1,\beta-1+i,n+1)\right) \\ &+(1-\mu)\left(\sum_{i=0}^{M-k-\beta} W(i)V(k,\beta+i,n+1)\right)\Bigg\}; \end{aligned} \tag{3.3}$$

If $\beta = 0$:

$$V(k,0,n) = \max\{R(k) - \lambda(n\tau + t_0), \sum_{i=0}^{M-k-\beta} W(i)V(k,i,n+1)\Big\}, \tag{3.4}$$

where $W(i) = \binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i}$.

Theorem 4. *The following threshold policy is optimal for Problem* (3.2)*: For any state* (k, β, n)*, policy* π^* *first check* β*.*

- *If* $\beta > 0$*: if* $k < k^*$*, continue; if* $k \geq k^*$*, stop;*
- *If* $\beta = 0$*: if* $k < k_0^*$*, continue; if* $k \geq k_0^*$*, stop.*

In the algorithm above, $k^* = \min\{k_1^*, M\}$*,* $k_1^* = \inf\{k : \Delta R(k) \leq \frac{\lambda\tau}{\mu}\}$*, and* $k_0^* \leq k^*$*.*

Note that k^* has a closed-form but k_0^* does not. We next discuss how to calculate k_0^* numerically based on the following lemma. The proof can be found in the appendix.

Lemma 1. *Given any* $\lambda > 0$*, and any state* (k, β, n) *such that* $k \leq k^*$*,* $k + \beta \geq k^*$ *and* $n \geq k$*, we have*

$$V(k,\beta,n) = R(k^*) - \lambda(n\tau + t_0) - (k - k^*)\frac{\lambda\tau}{\mu}.$$

From the proof of Theorem 4 we can omit n in state(k, β, n) and just need to calculate a two-dimensional value table of (k, β) with a fixed n. We can calculate this value using dynamic programming and setting a fixed n which is larger than M. This is a value table with size of $M \times M$. Value of states with different n can be calculated directly from Lemma 3: For any state (k, β, n) with $n \geq k$, $V(k, \beta, n+1) = V(k, \beta, n) - \lambda\tau$. (Proof can be found in appendix.)

Furthermore, the value for states with $k \geq k^*$and $n \geq k$, and it has been shown that the value for states with $k \leq k^*$, $k + \beta \geq k^*$ and $n \geq k$, so that we need to calculate a value table with size $k^* \times k^*$ instead of size $M \times M$, which are value of states with $k + \beta < k^*$. So we only need to use dynamic programming to calculate k_0^* as well as V_λ ($V(0, 0, 0)$).

We start from state $(k^* - 1, 0, n)$, where n is a fixed number larger than M. The Bellman equation is as follows:

$$V(k^* - 1, \beta = 0, n) = \max\{R(k^* - 1) - \lambda(n\tau + t_0), \sum_{i=0}^{M-k^*+1} G(i) \cdot V(k^* - 1, i, n) - \lambda\tau\} \tag{3.5}$$

where $G(i) = \binom{M-k^*+1}{i} p^i (1-p)^{M-k^*+1-i}$.
Since $V(k^*-1, i \geq 1, n)$ are known, and $V(k^*-1, i \geq 1, n) = R(k^*)-\lambda(n\tau+t_0)-\lambda\tau/\mu$ according to Lemma 1, so we can calculate this Bellman equation and get $V(k^*-1, i \geq 1, n)$.

Similarly we calculate the values of state $(k^* - 2, 1, n)$ and $(k^* - 2, 0, n)$. The optimal rule for state $(k^* - 2, 1, n)$ is to continue according to Lemma 3, so

$$\begin{aligned} V(k^* - 2, 1, n) = \mu &\left(\sum_{i=0}^{M-k^*+1} G(i) V(k^* - 1, i, n + 1) \right) \\ &+ (1 - \mu) \left(\sum_{i=0}^{M-k^*+1} G(i) V(k^* - 2, 1 + i, n) \right) - \lambda\tau, \end{aligned} \tag{3.6}$$

where $G(i) = \binom{M-k^*+1}{i} p^i (1-p)^{M-k^*+1-i}$. All values on the right hand side are known, so $V(k^* - 2, 1, n)$ is done.

For $(k^*-2,0,n)$, we have the following Bellman equation:

$$V(k^*-2,\beta=0,n) = \max\{R(k^*-2)-\lambda(n\tau+t_0), \sum_{i=0}^{M-k^*+2} H(i)\cdot V(k^*-2,i,n)-\lambda\tau\Big\}, \tag{3.7}$$

where $H(i) = \binom{M-k^*+2}{i}p^i(1-p)^{M-k^*+2-i}$. Since we just get $V(k^*-2,1,n)$, and $V(k^*-2,i\geq 2,n)$ are known by Lemma 1, after solving this bellman equation, we can get $V(k^*-2,\beta=0,n)$.

Next we continue calculate $V(k^*-3,2,n)$,$V(k^*-3,1,n)$, $V(k^*-3,0,n)$; and then $V(k^*-4,3,n)$,$V(k^*-4,2,n)$, $V(k^*-4,1,n)$,$V(k^*-4,0,n)$ $\cdots$, $V(0,\beta=k^*-1,n), V(0,\beta=k^*-2,n),...,V(0,\beta=0,n)$. Now we have got this $k^*\times k^*$ value table.

According to Lemma 3, we can get $V_\lambda = V(0,0,0) = V(0,0,n)+n\lambda\tau$. According to Lemma 6, we can calculate

$$k_0^* = \min\left\{M, \inf\{k : V(k,0,n) = R(k)-\lambda(n\tau+t_0)\}\right\},$$

which determines the optimal stopping rule according to Theorem 4.

The algorithm is summarized below for given $\lambda>0$.

Then we show a lower bound for J^* and an upper bound for k^*.

Corollary 1. *J^* is lower bounded by*

$$\frac{R(1)}{(\frac{1}{\mu}+\frac{1}{p})\tau+t_0}.$$

Proof. By definition of J^*, we choose a policy π which is to stop when $k=1$. so $J^*\geq \frac{R(1)}{\mathbb{E}_\pi(N\tau+t_0)}$. If $M=1$, then $\mathbb{E}_\pi(N)=\frac{1}{\mu}+\frac{1}{p}$, so for any $M\geq 1$, $\mathbb{E}_\pi(N)\leq \frac{1}{\mu}+\frac{1}{p}$, then $J^*\geq \frac{R(1)}{\mathbb{E}_\pi(N\tau+t_0)}\geq \frac{R(1)}{(\frac{1}{\mu}+\frac{1}{p})\tau+t_0}$.

□

Algorithm 1: Solution algorithm for problem 3.2

Given parameters: M,t_0,τ,λ, μ, p and reward function $R(k)$;

Calculate k^*, where $k^* = min\{M, k_1^*\}$, $k_1^* = inf\{k : \Delta R(k) \leq \frac{\lambda\tau}{\mu}\}$

Set $n = k_1^* + 1$;

Get $V(k, \beta, n) = R(k) - \lambda(n\tau + t_0)$ directly by lemma 2, where

$k \geq k^*, 0 \leq \beta \leq M - k^*$;

Get $V(k, \beta, n) = R(k^*) - \lambda(n\tau + t_0) - (k^* - k)\lambda\tau/\mu$ by lemma 5, where

$k < k^*, k^* - k \leq \beta \leq M - k$;

Calculate $V(k^* - 1, \beta = 0, n)$ by solving bellman equation;

Calculate $V(k^* - 2, \beta = 1, n), V(k^* - 2, \beta = 0, n)$;

$V(k^* - 3, \beta = 2, n), V(k^* - 3, \beta = 1, n),$

$V(k^* - 3, \beta = 0, n)$;

...

$V(0, \beta = k^* - 1, n), V(0, \beta = k^* - 2, n),$

$..., V(0, \beta = 0, n)$ sequentially;

Return optimal value $V_\lambda = V(0, 0, 0) = V(0, 0, n) + \lambda n$;

Calculate $k_0^* = inf\{k : V(k, 0, n) = R(k) - \lambda(n\tau + t_0)\}$,

Return optimal policy π_λ by theorem 2.

If $\beta > 0$: If $k < k^*$, continue; else, stop

else: If $k < k_0^*$: continue; else, stop.

Corollary 2. *J^* is upper bounded by $\max_k \frac{R(k)}{k\tau+t_0}$, where k is an integer and $0 \leq k \leq M$.*

Proof. An obvious upper bound for J^* is $J' = \sup_{\pi \in S} \frac{\mathbb{E}[R(K_N)]}{\mathbb{E}[N \cdot \tau + t_0]}$, where p and μ are both set to 1, so J^* is upper bounded by $\max_k \frac{R(k)}{k\tau+t_0}$, where k is integer and $0 \leq k \leq M$. □

3.3 The solution of the Original Problem

In the previous section, we have shown how to find optimal π_λ for problem 3.2 as well as V_λ for a given $\lambda > 0$.

Next we focus on finding the optimal λ so that we can solve the original problem. Corollary 1 shows a lower bound on J^* , we can use this to initialize the λ. The following lemma demonstrates some important properties of V_λ.

Lemma 2. *V_λ is decreasing and convex in λ.*

Proof. The idea of proof comes from Ferguson (2012). Assume $\lambda_1 < \lambda_2$, then

$$\begin{aligned} V_{\lambda_2} &= E_{\pi_{\lambda_2}}[R(K_N) - \lambda_2(N\tau + t_0)] \\ &< E_{\pi_{\lambda_2}}[R(K_N) - \lambda_1(N\tau + t_0)] \\ &\leq E_{\pi_{\lambda_1}}[R(K_N) - \lambda_1(N\tau + t_0)] \\ &= V_{\lambda_1}, \end{aligned}$$

so V_λ is decreasing with λ.

To prove convexity, given λ_1 and λ_2, let $0 < \theta < 1$, $\lambda = \theta\lambda_1 + (1-\theta)\lambda_2$, so

$$\begin{aligned} V_\lambda &= E_{\pi_\lambda}[R(K_N) - (\theta\lambda_1 + (1-\theta)\lambda_2)(N\tau + t_0)] \\ &= \theta E_{\pi_\lambda}[R(K_N) - \lambda_1(N\tau + t_0)] \\ &\quad + (1-\theta)E_{\pi_\lambda}[R(K_N) - \lambda_2(N\tau + t_0)] \\ &\leq \theta V_{\lambda_1} + (1-\theta)V_{\lambda_2}. \end{aligned}$$

□

Since we initialize λ using the lower bound in Corollary 1, we can get a upper bound on k^* for all possible λ when $R(k) = c - \frac{a}{k}$ for some constants a and c.

Corollary 3. *Given* $R(k) = c - \frac{a}{k+1}$, *we have* $k^* < \sqrt{\frac{a}{c-a/2}(1 + \frac{\mu}{p} + \frac{\mu t_0}{\tau})}$.

Proof. It is directly from the definition of k^* and Corollary 1. Since $k^* = \inf\{k : \Delta R(k) \leq \frac{\lambda\tau}{\mu}\}$, we have

$$\frac{a}{k^*(k^*+1)} > \lambda\tau/\mu > \frac{c - a/2}{(\frac{1}{\mu} + \frac{1}{p})\tau + t_0}\tau/\mu.$$

Therefore,

$$k^* < \sqrt{\frac{a}{c - a/2}(1 + \frac{\mu}{p} + \frac{\mu t_0}{\tau})}.$$

□

Corollary 3 obtains an upper bound on k^*, which depends on parameters a, c, μ, p, t_0, and τ, but independent of M. Therefore, the complexity of the algorithm for (3.2) is $O(M(k^*)^2) = O(M)$.

Next we show details for solving problem 3.1. In this algorithm, σ is the predefined accuracy level.

3.4 Evaluation

Data and Model: We consider the experiment of training a CNN model with distributed MNIST data. The dataset is divided into 100 groups, each representing a local dataset (or a client). Each client trains the CNN model with its own data and uploads its newly trained parameters sequentially to a parameter server if the channel is ON.

Reward function: We first plot the reward function $R(k)$ which is defined to be the decrement of the loss function (the cross-entropy loss) when the number of

Algorithm 2: Solution for problem 1

Given parameters: M,t_0,τ, μ, p ; Reward function R(k);

Step 1: Calculate lower bound λ_{lower} and upper bound λ_{upper} by corollary 1 and corollary 2.

Step 2: $\lambda = (\lambda_{upper} + \lambda_{lower})/2$

while $|V_\lambda| > \sigma = 0.001$ **do**

  if $V_\lambda > 0$ **then**

    $\lambda_{lower} \leftarrow \lambda$, $\lambda \leftarrow (\lambda_{upper} + \lambda_{lower})/2$

  else

    $\lambda_{upper} \leftarrow \lambda$, $\lambda \leftarrow (\lambda_{upper} + \lambda_{lower})/2$

  end

end

Step 4: return λ, which is equal to J^*

updates increases from $k-1$ to k. The loss function for $k = 1, \cdots, 40, ...$ is shown in Fig. 3.2, from which we can see that $R(k) = c - \frac{a}{k+1}$ fits the reward function well. So in our experiments, we assume $R(k) = c - \frac{a}{k+1}$ for some $a > 0$ and $c > 0$.

We evaluated the proposed algorithms using the MNIST dataset. In our experiment, we chose $M = 100$, $t_0 = 3,000$ms which includes broadcasting time and aggregating time, $\tau = 10$ms, and defined the reward function to be $R(k) = 0.04 - \frac{0.018}{k+1}$.

3.4.1 Geometric Distribution Case

About parameters transmissions μ and success probability of data processing p in the model, we further chose success probability of transmissions μ to be $\frac{5}{8}$. The size of the parameters of our CNN is about 80k-100k. Based on the transmission rate of current 4G systems, which is about 50k per 10ms, we assume the average time for finishing uploading the parameters is around 16ms, which leads to our choice of

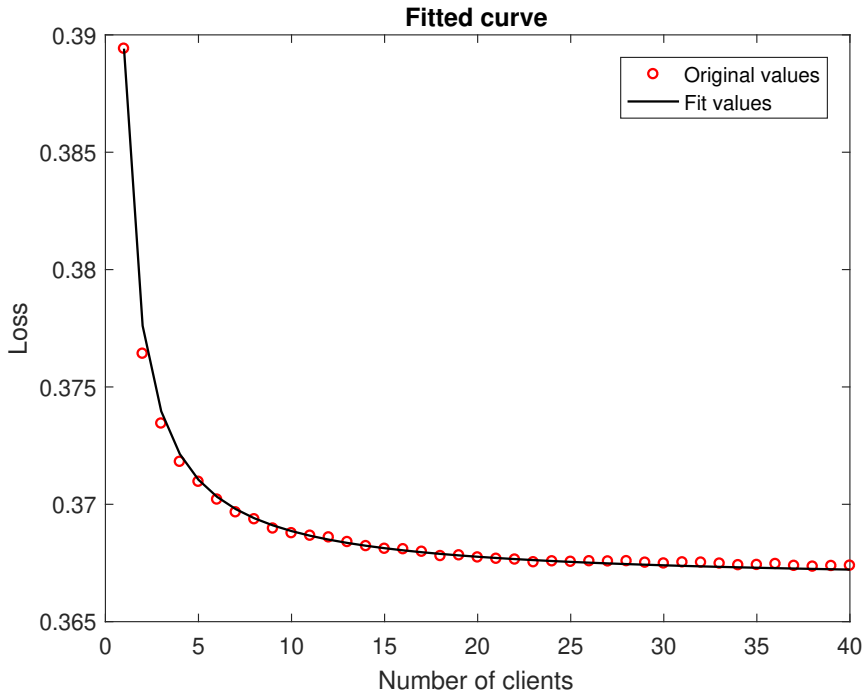

Figure 3.2: Simulation Result about Reward Function(Loss Function) with 100 User

μ. The success probability of data processing p is set to be $1/500$. We estimate the average training time on cell phone is $5000ms$. Since the duration of each time slot is $10ms$, the transmission probability is set to be $1/500$.

We compared the loss function under the proposed algorithm based on optimal stopping time and other algorithms based on fixed number of updates in each round. In particular, we considered two other algorithms: the first algorithm require updates from all M devices and the second algorithm, used by Google, requires 10% updates from the M device. We remark that the 10% rule is selected by comparing different fractions and found the best one for each application McMahan *et al.* (2016). So it can be viewed as a policy uses the "optimal" number of updates at each iteration.

For the optimal stopping time algorithm, we first obtain lower bound $\lambda_{\text{lower}} = 0.01203$ and upper bound $\lambda_{\text{upper}} = 0.01234$. We then found the optimal $\lambda^* = 0.01209$,

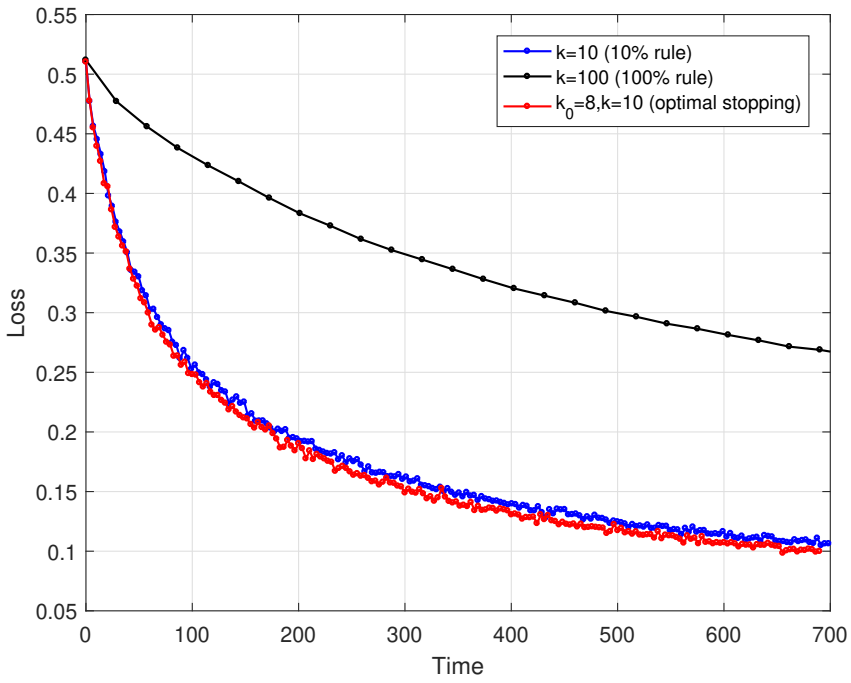

Figure 3.3: Experiment Result Using Optimal Stopping Rule with 100 Users

from which, we obtained that k^*=10 and $k_0^* = 8$ for the optimal stopping rule.

The testing loss as a function time is shown in Figure 3.3. Each data point in lines represents the test loss after one round of training. The length interval between two data points in each line of our figure shows average running time in one round for each stopping rule. For example, the average simulated running time of a round with the optimal stopping rule ($k^* = 10$, $k_0^* = 8$) is 3.45s, as well as $3.55s$ for $k = 10$ and 28.76s for $k = 100$. We can see that the optimal stopping rule reduce the loss by at least 170% throughout the training process. Figure 3.4 compares just the optimal stopping rule and the 10% rule. We again can see from this figure that even comparing to the "optimal" fixed k, the optimal stopping rule still reduces the loss function by 7% throughout.

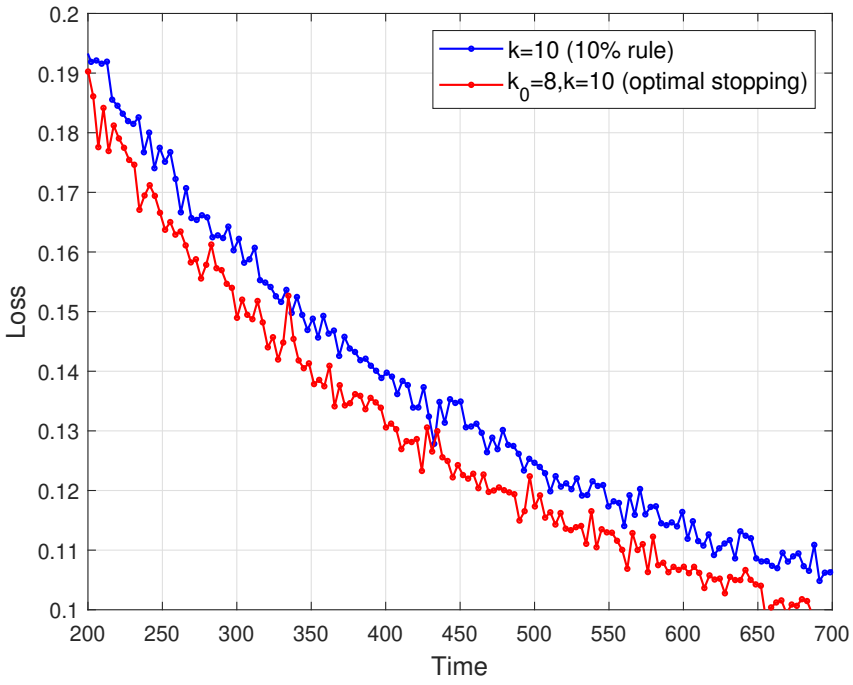

Figure 3.4: Experiment Result Using Optimal Stopping Rule with 100 Users

3.4.2 Heavy-Tailed Case

In real life, Heavy-tailed Distributions are more general. In this subsection of evaluation, instead of assuming that transmission and computation process follows geometry distribution, we assume transmission and computation process are i.i.d and follow Pareto distribution. Now we introduce how we choose parameters of Pareto distribution. In last subsection, we assume average time for finishing uploading the parameters is around 16ms, which is 1.6 time slot. This leads to our choice of k, m, which is $k = 1.454, m = 0.5$. Also We estimate average training time on cell phone is $5000ms$, which is 500 time slots, so we choose k, m for computing process is $k = 2, m = 250$.

All other models and data are the same as them in last subsection. Now we will show how well our algorithm works if real transmission and computation process

does not follow geometric distribution but heavy-tailed(Pareto distribution). Just as in last geometric case, We also compared the loss function under the proposed algorithm based on optimal stopping time and other algorithms based on fixed number of updates in each round. The 10% rule is also selected as our bench mark.

Since all data and models are same, k^*=10 and $k_0^* = 8$ is still our optimal stopping rule in this scenario.

The testing loss as a function time is shown in Figure 3.5. The average simulated running time of a round with the optimal stopping rule ($k^* = 10$, $k_0^* = 8$) is 5.7s, as well as $5.71s$ for $k = 10$ and 46.56s for $k = 100$. We can see that the optimal stopping rule still reduces the loss by at least 130% throughout the training process even transition and computation process follow heavy-tailed distribution. Figure 3.6 compares just the optimal stopping rule and the 10% rule. We again can see from this figure that even comparing to the "optimal" fixed k, the optimal stopping rule still reduces the loss function by 4.5% throughout.

This evaluation result shows that our optimal rule given by our algorithm works very well even the computing and transmission process does not follow geometric distribution which is assumed in our model. If computing and transmission process follow some heavy tailed distribution, our algorithm can still provide a very promising stopping rule on problem when to optimally stop a round in federated learning.

In the following sections, we introduce the proof of Theorem 3, theorem 4 and Lemma 1.

3.5 Proof of Theorem 3

Proof. Given some λ, if $\sup_{\pi \in S} \mathbb{E}(R(K_N) - \lambda(N\tau + t_0)) = 0$, then $\mathbb{E}(R(K_N) - \lambda(N\tau + t_0)) \leq 0$ for all π, so $\mathbb{E}\, R(K_N)/\, \mathbb{E}(N\tau + t_0) \leq \lambda$ under all π, then $\sup_{\pi \in S} \mathbb{E}\, R(K_N)/\, \mathbb{E}(N\tau + t_0) \leq \lambda$.

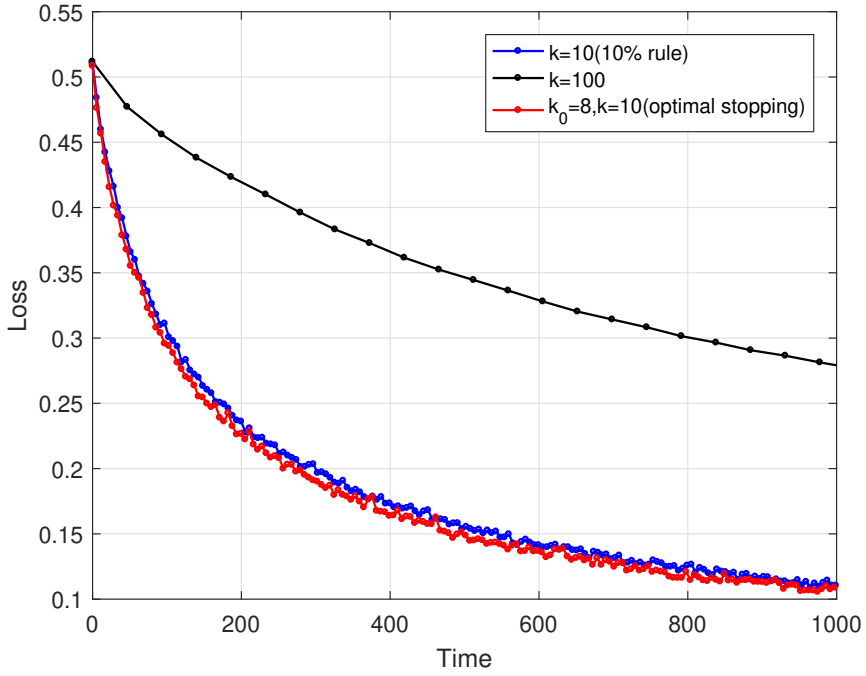

Figure 3.5: Experiment Result Using Optimal Stopping Rule with 100 Users in Heavy-tailed Case

If $V_\lambda = \sup_{\pi \in S} \mathbb{E}(R(K_N) - \lambda(N\tau + t_0)) = 0$ and is attained at some policy $\pi^* \in S$, then for any positive ϵ, $\mathbb{E}_{\pi^*}(R(K_N) - \lambda(N\tau + t_0)) \geq -\epsilon$, then $\mathbb{E}_{\pi^*}(R(K_N)/\,\mathbb{E}_{\pi^*}(N\tau + t_0) \geq \lambda - \epsilon/\,\mathbb{E}_{\pi^*}(N\tau + t_0) \geq \lambda - \epsilon/t_0$, since ϵ is positive and arbitrary small, so $\sup_{\pi \in S} \mathbb{E}\, R(K_N)/\,\mathbb{E}(N\tau + t_0) \geq \lambda$.

So we have $J^* = \sup_{\pi \in S} \mathbb{E}\, R(K_N)/\,\mathbb{E}(N\tau + t_0) = \lambda$.

Moreover, if $V_\lambda = \mathbb{E}_{\pi^*}(R(K_N) - \lambda(N\tau + t_0)) = 0$, then $\mathbb{E}_{\pi^*} R(K_N^*)/\,\mathbb{E}_{\pi^*}(N^*\tau + t_0) = \lambda = \sup_{\pi \in S} \mathbb{E}\, R(K_N)/\,\mathbb{E}(N\tau + t_0)$, so π^* is an optimal policy for maximizing $\mathbb{E}\, R(K_N)/\,\mathbb{E}(N\tau + t_0)$. □

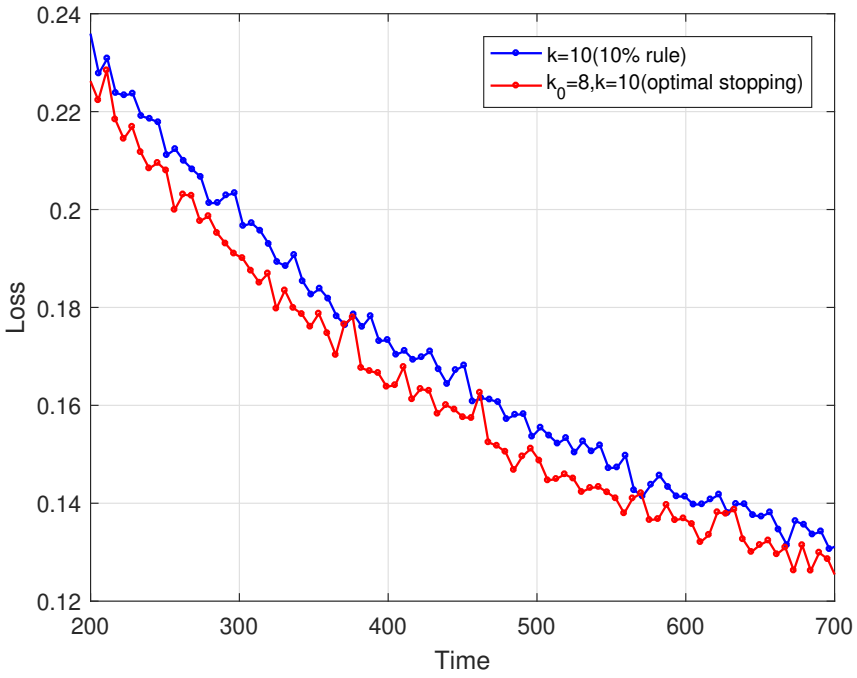

Figure 3.6: Experiment Result Using Optimal Stopping Rule with 100 Users in Heavy-tailed Case

3.6 Proof of Theorem 4

Before the proof of theorem 4, we first introduce some lemmas. Remark that for any possible state (k, β, n), $n \geq k$ always holds, so we only need to consider states with $n \geq k$.

Lemma 3. *For any state (k, β, n) with $n \geq k$, $V(k, \beta, n+1) = V(k, \beta, n) - \lambda\tau$.*

Proof. The proof directly comes from the definition of $V(k, \beta, n)$,

$$\begin{aligned} V(k, \beta, n+1) &= \sup_{\pi} \mathbb{E}\left[R(k + K_N) - \lambda((N + n + 1)\tau + t_0)\right] \\ &= \sup_{\pi} \mathbb{E}\left[R(k + K_N) - \lambda((N + n)\tau + t_0)\right] - \lambda\tau \\ &= V(k, \beta, n) - \lambda\tau \end{aligned}$$

□

By the principle of optimality, an optimal policy is to compare $V(k,\beta,n)$ with $R(k)-\lambda(n\tau+t_0)$, which is the reward of stopping at state (k,β,n). If $V(k,\beta,n)=R(k)-\lambda(n\tau+t_0)$, then stop; if $V(k,\beta,n)>R(k)-\lambda(n\tau+t_0)$, then continue. Lemma 1 shows that the optimal actions are same for any two states (k,β,n) and (k,β,n') with $n,n'\geq k$. Because if $V(k,\beta,n)=R(k)-\lambda(n\tau+t_0)$ for some $n\geq k$, which means to stop is optimal, then $V(k,\beta,n')=R(k)-\lambda(n'\tau+t_0)$ for any $n'\geq k$, so to stop is also optimal for any state $V(k,\beta,n')$; if $V(k,\beta,n)>R(k)-\lambda(n\tau+t_0)$ for some $n\geq k$, which means to continue is optimal, then $V(k,\beta,n')>R(k)-\lambda(n'\tau+t_0)$ for any $n'\geq k$, so continue is optimal for any state $V(k,\beta,n')$.

Lemma 4. *For all states with $k\geq k^*$ and $n\geq k$, optimal action rule is to stop, and $V(k,\beta,n)=R(k)-\lambda(n\tau+t_0)$.*

Proof. If $k^*=M$, then to stop is naturally.

Now we consider $k^*<M$. The idea of proof is from the conclusion in Ferguson (2012) that one-stage-look-ahead rule (1-sla rule) is optimal for monotone problem if problem is in finite horizon or problem is in infinite horizon, but can be approximated well by finite horizon problem in the sense that $V_0^J\to V_0^\infty$ as $J\to\infty$, where V_0^J denotes the optimal return for the problem truncated at terminal stage J, and V_0^∞ denotes the optimal return for the original infinite horizon problem.

Now we first give the definition of monotone problem. Let $Z_n=R(K_n)-\lambda(n\tau+t_0)$, and let A_n denote events $\{Z_n\geq\mathbb{E}(Z_{n+1}|(X_n,n))\}$, A_{n+1} denote events $\{Z_{n+1}\geq\mathbb{E}(Z_{n+2}|(X_{n+1},n+1))\}$. The set A_n is the set of all states (k,β,n) on which the 1-sla calls for stopping at some fixed n. If $A_n\subseteq A_{n+1}\subseteq A_{n+2}...$, then this problem is monotone. $A_n\subseteq A_{n+1}\subseteq A_{n+2}\subseteq ...$ means that if the 1-sla calls for stopping at stage n, then it will call for stopping at all future stages no matter what the future

observations turn out to be.

Then we show problem 3.2 is monotone when $k \geq k^*$.

It is obvious that for any state $(k, 0, n)$ with $k \geq k^*$, $R(k) - \lambda(n\tau + t_0) > R(k) - \lambda((n+1)\tau + t_0) = \mathbb{E}(Z_{n+1}|(X_n, n) = (k, \beta, n))$, which means 1-sla rule will always calls for stop for any state $(k, 0, n)$ with $k \geq k^*$.

For any state $(k, \beta > 0, n)$ with $k \geq k^*$, it can be seen that

$$\begin{aligned} & R(k) - \lambda(n\tau + t_0) - \mu[R(k+1) - \lambda((n+1)\tau + t_0)] \\ & \quad - (1-\mu)[R(k) - \lambda((n+1)\tau + t_0)] \\ =& \lambda\tau - \mu\Delta R(k) \geq 0, \end{aligned} \tag{3.8}$$

which means

$$\begin{aligned} & R(k) - \lambda(n\tau + t_0) \\ \geq & \mu[R(k+1) - \lambda((n+1)\tau + t_0)] \\ & \quad + (1-\mu)[R(k) - \lambda((n+1)\tau + t_0)] \\ = & \mathbb{E}(Z_{n+1}|(X_n, n) = (k, \beta, n)) \end{aligned}$$

always holds for all states with $k \geq k^*$, which means 1-sla rule will always calls for stop for any state $(k, \beta > 0, n)$ with $k \geq k^*$.

So our problem is monotone for any state with $k \geq k^*$.

Since Our problem is in infinite horizon, and can be approximated well by finite horizon, that is because $V_\lambda^J \to V_\lambda$ as $J \to \infty$ in our problem, and V_λ^J denotes optimal value of problem 3.2 truncated at terminal stage J, so 1-sla rule is optimal when $k \geq k^*$. Since one stage ahead rule tells to stop for all states with $k \geq k^*$, so to stop at all states with $k \geq k^*$ are optimal. □

Lemma 5. *For all states with $k < k^*$, $n \geq k$ and $\beta \geq 1$, optimal rules are to continue.*

Proof. For any state (k, β, n) with $k < k^*$ and $\beta \geq 1$, the reward to stop is $R(k) - \lambda(n\tau + t_0)$, the expected reward to continue one more stage and then stop is $\mu[R(k+1) - \lambda((n+1)\tau + t_0)] + (1-\mu)[R(K) - \lambda((n+1)\tau + t_0)]$.

We have

$$\begin{aligned} & R(k) - \lambda((n)\tau + t_0) - \mu[R(K+1) - \lambda((n+1)\tau + t_0)] \\ & \quad - (1-\mu)[R(K) - \lambda((n+1)\tau + t_0)] \\ =& \lambda\tau - \mu\Delta R(k), \end{aligned}$$

By the definition of k^*, for all $k < k^*$ and $\beta \geq 1$, $\lambda\tau - \mu\Delta R(k) < 0$. So for all states with $k < k^*$ and $\beta \geq 1$, to continue is always better than to stop.

□

Then we deal with states with $k < k^*$ and $\beta = 0$, and we have following lemma:

Lemma 6. *For all states with $k < k^*$, $\beta = 0$ and $n \geq k$, if optimal rule for state $(k, 0, n)$ is to continue, then optimal rule for $(k-1, 0, n)$ is also to continue.*

Proof. Assume optimal rule for $(k, 0, n)$ is to continue, then

$$\begin{aligned} V(k, 0, n) &= \sup_{\pi} \mathbb{E}[R(k + K_N) - \lambda((n+N)\tau + t_0)] \\ &> R(k) - \lambda(n\tau + t_0), \end{aligned}$$

where $K_N \leq M - k$. And we move the right hand part to the left, then

$$\sup_{\pi: K_N \leq M-k} \mathbb{E}[R(k + K_N) - R(k) - \lambda(N\tau)] > 0$$

The left hand size $\sup_\pi \mathbb{E}[R(k + K_N) - R(k) - \lambda(N\tau)]$ can be thought as optimal value of a new stopping problem starting from state $(k, 0, 0)$, with reward of getting $K_N + k$ clients' parameters at stage N to be $R(k + K_N) - R(k) - \lambda N\tau$ (if choosing to

stop) and the number of unfinished clients in this problem is $M-k$, so $K_N \leq M-k$, . Since $R(k+K_N)-R(k) \leq R(k-1+K_N)-R(k-1)$, by assumption about $R(k)$, we have

$$\sup_{\pi:K_N \leq M-k} \mathbb{E}(R(k-1+K_N)-R(k-1)-\lambda(N\tau) > 0$$

The left hand size also can be thought as the optimal value of a stopping problem starting from state$(k-1,0,n)$, but with reward of getting K_N+k-1 clients' parameters to be $R(k-1+K_N)-R(k-1)-\lambda N\tau$ and $K_N \leq M-k$, the number of unfinished clients in this problem is $M-k$, we use $V'(k-1,0,n)$ to denote this value. Now we consider a similar stopping problem also starting from state$(k-1,0,n)$ and same reward $R(k-1+K_N)-R(k-1)-\lambda N\tau$, what is different is now the the number of unfinished clients at this new problem increases 1 to $M-k+1$, so now $K_N \leq M-k+1$. Let $V''(k-1,0,n)$ denotes optimal value of this new problem. It is obvious that $0 < V'(k-1,0,n)) \leq V''(k-1,0,n))$, since larger numbers of clients will decrease the possibility of state with $\beta = 0$ and so leads to larger optimal value. So

$$\begin{aligned} V''(k-1,0,n) = & \sup_{\pi:K_N \leq M-k+1} \mathbb{E}(R(k-1+K_N)) \\ & - R(k-1) - \lambda(N\tau) \\ > & \ 0 \end{aligned}$$

Then

$$\begin{aligned} & \sup_{\pi:K_N \leq M-k+1} \mathbb{E}[R(k-1+K_N) - \lambda((n+N)\tau + t_0)] \\ & > R(k-1) - \lambda(n\tau + t_0) \end{aligned}$$

The left hand side is the value of state $(k-1,0,n)$ in our problem, right hand side is the reward of stopping at this state, So optimal rule for state $(k-1,0,n)$ is also to continue. □

By proof of lemma 3, 4, 5 and 6, we can directly get theorem 4.

3.7 Proof of Lemma 1

Proof. If $k = k^*$, then $\beta = 0$ and by lemma 4, optimal rule is to stop now, and $V(k^*, 0, n) = R(k^*) - \lambda \cdot (n\tau + t_0)$.

If $k < k^*$, then by $k + \beta \geq k^*$, we have $\beta > 0$, and by lemma 3, optimal action for all states with $k < k^*$ and $\beta > 0$ is to continue, So we write down the bellman equation for any state (k, β, n) with $k < k^*$, $k + \beta \geq k^*$ and $n \geq k$:

$$\begin{aligned} V(k,\beta,n) &= \mu[\sum_{i=0}^{M-k-\beta} L(i) \cdot V(k+1, \beta-1+i, n+1)] \\ &\quad + (1-\mu)[\sum_{i=0}^{M-k-\beta} L(i) \cdot V(k, \beta+i, n+1)] \qquad (3.9) \\ &= \mu[\sum_{i=0}^{M-k-\beta} L(i) \cdot V(k+1, \beta-1+i, n)] \\ &\quad + (1-\mu)[\sum_{i=0}^{M-k-\beta} L(i) \cdot V(k, \beta+i, n)] - \lambda\tau, \qquad (3.10) \end{aligned}$$

where $L(i) = \binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i}$. Last equation comes from lemma 3.

We first start from $\beta = M - k$, then bellman equation becomes:

$$\begin{aligned} V(k,\beta,n) &= V(k, M-k, n) \\ &= \mu V(k+1, M-k-1, n) \\ &\quad + (1-\mu)V(k, M-k, n) - \lambda\tau, \qquad (3.11) \end{aligned}$$

so

$$V(k, M-k, n) = V(k+1, M-k-1, n) - \lambda\tau/\mu, \qquad (3.12)$$

Also we have

$$V(k+1, M-k-1, n) = V(k+2, M-k-2, n) - \lambda\tau/\mu,$$

$$...$$

so

$$\begin{aligned} & V(k, M-k, n) \\ &= V(k+1, M-k-1, n) - \lambda\tau/\mu \qquad (3.13) \\ &= V(k+2, M-k-2, n) - \lambda\tau/\mu - \lambda\tau/\mu \\ & ... \\ &= V(k+k^*-k, M-k^*, n) - (k^*-k)\lambda\tau/\mu \\ &= R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu \qquad (3.14) \end{aligned}$$

.

We continue check $\beta = M-k-1$, $\beta = M-k-2$, $\beta = M-k-3$,..., , until $\beta = k^*-k$. We first consider state (k, β, n) with $\beta = M-k-1$, and the bellman equation now is:

$$\begin{aligned} & V(k, M-k-1, n) \\ =& \mu[pV(k+1, M-k-1, n) \\ & + (1-p)V(k+1, M-k-2, n)] \\ & + (1-\mu)[pV(k, M-k, n) \\ & + (1-p)V(k, M-k-1), n] - \lambda\tau \qquad (3.15) \end{aligned}$$

If we Move last part in the right hand side to left and by equation $V(k, M-k, n) =$

$V(k+1, M-k-1, n) - \lambda\tau/\mu$, we have

$$
\begin{aligned}
&(p+\mu-\mu p)V(k, M-k-1, n) \\
=&pV(k, M-k, n) + \mu(1-p)V(k+1, M-k-2), n] - (1-p)\lambda\tau \qquad (3.16)\\
=&p[R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu] \\
&+ \mu(1-p)V(k+1, M-k-2), n] - (1-p)\lambda\tau. \qquad (3.17)
\end{aligned}
$$

so it's easy to check that

$$V(k, M-k-1, n) = R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu \qquad (3.18)$$

satisfies the bellman equation, because if it holds, then
$V(k, M-k-1, n) = V(k, M-k, n)$ and $V(k, M-k-1, n) = V(k+1, M-k-2, n) - \lambda\tau/\mu$, then it is obvious that bellman equation holds.

Now we show if $V(k, \beta', n) = R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu$ for all states with $k + \beta' \geq M - m$, where $m \leq M - k^* - 1$, then $V(k, \beta, n) = R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu$ with $\beta = M - k - m - 1$, then by induction, we can get our result.

We first write down the bellman equation:

$$
\begin{aligned}
&V(k, \beta, n) \\
=&\mu[\sum_{i=0}^{M-k-\beta} \binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i} \cdot V(k+1, \beta-1+i, n)] \\
&+ (1-\mu)[\sum_{i=0}^{M-k-\beta} \binom{M-k-\beta}{i} p^i (1-p)^{M-k-\beta-i} \cdot V(k, \beta+i, n)] \\
&- \lambda\tau,
\end{aligned}
$$

Since for any $i \geq 1$, $k + \beta + i = M - m - 1 + i \geq M - m$, so $V(k+1, \beta-1+i, n) = R(k^*) - \lambda(n\tau + t_0) - (k^* - k - 1)\lambda\tau/\mu$, $V(k, \beta+i, n) = R(k^*) - \lambda(n\tau + t_0) - (k^*-k)\lambda\tau/\mu$,

So

$$
\begin{aligned}
&(1-(1-\mu)(1-p)^{M-k-\beta})V(k,\beta,n)\\
=&\mu(1-p)^{M-k-\beta}V(k+1,\beta-1,n)\\
&+(1-(1-p)^{M-k-\beta})[R(k^*)-\lambda(n\tau+t_0)-(k^*-k)\lambda\tau/\mu]\\
&-(1-p)^{M-k-\beta}\lambda\tau,
\end{aligned}
$$

and we will see $V(k,\beta,n) = R(k^*)-\lambda(n\tau+t_0)-(k^*-k)\lambda\tau/\mu$ satisfies the bellman equation. So by induction, our proof is done. And condition $k+\beta \geq k^*$ is to guarantee that when $k = k^*$, $\beta \geq 0$. □

3.8 Summary

In this chapter, we studied the problem of when to optimally stop a round in federated learning. We formulate the problem as an optimal stopping problem and develop a low complexity algorithm to solve the stopping rule. Our experiments on real data set shows the significant improvements compared with policies that collect a fixed number of updates in each iteration.

Chapter 4

CONCLUSION

This dissertation focused on two realated but different field in machine learning, (1) how to collect private data, (2) how to run learning on distributed data (without collecting private data).In the first half of this dissertation (Chapter 2), we studied incentive mechanisms for private discrete distribution estimation. We first derived a lower bound on the minimum payment required for guaranteeing quality level, and then proposed WINTALL — a novel incentive mechanism. The expected payment of WINTALL matches the lower bound when the underlying parameter θ can be estimated by the platform accurately. We present its application to private discrete distribution estimation, where WINTALL rewards individuals whose reported answers match the most popular one. We also presented real-world experiments on Amazon Mechanical Turk to validate the novelty of WINTALL-inspired mechanisms.

Chapter 3 studied the problem about how to optimally stop one round in Federated Learning. We consider a system with a single parameter server (PS) and M client devices for training a predictive learning model with distributed data sets on the client devices. The clients communicate with the parameter server using a common wireless channel so each time, only one device can transmit. The training is an iterative process consisting of multiple rounds. At beginning of each round, each client trains the model, broadcast by the parameter server at the beginning of the round, with its own data. After finishing training, the device transmits the update to the parameter server when the wireless channel is available. The server aggregates updates to obtain a new model and broadcasts it to all clients to start a new round. We formulate the problem that server decides when to stop/restart a new round as

an optimal stopping problem. Then we develop a low complexity algorithm to solve the modified problem, which also works for the original problem. Experiments on a real data set shows significant improvements compared with policies collecting a fixed number of updates in each iteration.

REFERENCES

Apple `https://images.apple.com/privacy/docs/Differential_Privacy_Overview.pdf` (2020).

Cai, Y., C. Daskalakis and C. Papadimitriou, "Optimum statistical estimation with strategic data sources", in "Proc. Conf. Learning Theory (COLT)", pp. 280–296 (2015).

Chen, M., R. Mathews, T. Ouyang and F. Beaufays, "Federated learning of out-of-vocabulary words", arXiv preprint arXiv:1903.10635 (2019).

Cummings, R., K. Ligett, A. Roth, Z. S. Wu and J. Ziani, "Accuracy for sale: Aggregating data with a variance constraint", in "Proc. Conf. Innovations in Theoretical Computer Science", pp. 317–324 (2015).

Dasgupta, A. and A. Ghosh, "Crowdsourced judgement elicitation with endogenous proficiency", in "Proc. Int. Conf. World Wide Web (WWW)", pp. 319–330 (2013).

Erlingsson, Ú., V. Pihur and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response", in "Proc. ACM Conf. Computer and Communications Security (CCS)", pp. 1054–1067 (Scottsdale, AZ, 2014).

Ferguson, T. S., "Optimal stopping and applications", (2012).

Fleischer, L. K. and Y. Lyu, "Approximately optimal auctions for selling privacy when costs are correlated with data", in "Proc. ACM Conf. Electronic Commerce (EC)", pp. 568–585 (Valencia, Spain, 2012).

Ghosh, A. and K. Ligett, "Privacy and coordination: Computing on databases with endogenous participation", in "Proc. ACM Conf. Electronic Commerce (EC)", pp. 543–560 (Philadelphia, PA, 2013).

Ghosh, A., K. Ligett, A. Roth and G. Schoenebeck, "Buying private data without verification", in "Proc. ACM Conf. Economics and Computation (EC)", pp. 931–948 (Palo Alto, CA, 2014).

Ghosh, A. and A. Roth, "Selling privacy at auction", in "Proc. ACM Conf. Electronic Commerce (EC)", pp. 199–208 (San Jose, CA, 2011).

Gong, X. and N. Shroff, "Incentivizing truthful data quality for quality-aware mobile data crowdsourcing", in "Proc. ACM Int. Symp. Mobile Ad Hoc Networking and Computing (MobiHoc)", pp. 161–170 (2018).

Hard, A., K. Rao, R. Mathews, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon and D. Ramage, "Federated learning for mobile keyboard prediction", arXiv preprint arXiv:1811.03604 (2018).

Kairouz, P., K. Bonawitz and D. Ramage, "Discrete distribution estimation under local privacy", in "Int. Conf. Machine Learning (ICML)", pp. 2436–2444 (2016).

Khetan, A. and S. Oh, "Achieving budget-optimality with adaptive schemes in crowdsourcing", in "Advances Neural Information Processing Systems (NIPS)", pp. 4844–4852 (2016).

Ligett, K. and A. Roth, "Take it or leave it: Running a survey when privacy comes at a cost", in "Proc. Int. Workshop Internet and Network Economics (WINE)", pp. 378–391 (Liverpool, UK, 2012).

Liu, Y. and M. Liu, "An online learning approach to improving the quality of crowdsourcing", in "Proc. Ann. ACM SIGMETRICS Conf.", pp. 217–230 (New York, NY, USA, 2015).

McMahan, H. B., E. Moore, D. Ramage, S. Hampson *et al.*, "Communication-efficient learning of deep networks from decentralized data", arXiv preprint arXiv:1602.05629 (2016).

Miller, N., P. Resnick and R. Zeckhauser, "Eliciting informative feedback: The peer-prediction method", in "Computing with Social Trust", Human–Computer Interaction Series, pp. 185–212 (Springer London, 2009).

Nissim, K., S. Vadhan and D. Xiao, "Redrawing the boundaries on purchasing data from privacy-sensitive individuals", in "Proc. Conf. Innovations in Theoretical Computer Science (ITCS)", pp. 411–422 (Princeton, NJ, 2014).

Prelec, D., "A bayesian truth serum for subjective data", Science **306**, 5695, 462–466 (2004).

Radanovic, G. and B. Faltings, "A robust bayesian truth serum for non-binary signals", in "AAAI Conf. Artificial Intelligence", pp. 833–839 (2013).

Radanovic, G. and B. Faltings, "Incentives for truthful information elicitation of continuous signals", in "AAAI Conf. Artificial Intelligence", pp. 770–776 (2014).

Radanovic, G. and B. Faltings, "Incentive schemes for participatory sensing", in "Proc. Int. Conf. Autonomous Agents and Multiagent Systems", pp. 1081–1089 (International Foundation for Autonomous Agents and Multiagent Systems, 2015).

Roth, A. and G. Schoenebeck, "Conducting truthful surveys, cheaply", in "Proc. ACM Conf. Electronic Commerce (EC)", pp. 826–843 (Valencia, Spain, 2012).

Shah, N. and D. Zhou, "No oops, you won't do it again: Mechanisms for self-correction in crowdsourcing", in "International Conference on Machine Learning", pp. 1–10 (2016).

Shah, N., D. Zhou and Y. Peres, "Approval voting and incentives in crowdsourcing", in "International Conference on Machine Learning", pp. 10–19 (2015).

Shah, N. B. and D. Zhou, "Double or nothing: Multiplicative incentive mechanisms for crowdsourcing", in "Advances in neural information processing systems", pp. 1–9 (2015).

Shnayder, V., A. Agarwal, R. Frongillo and D. C. Parkes, "Informed truthfulness in multi-task peer prediction", in "Proceedings of the 2016 ACM Conference on Economics and Computation", pp. 179–196 (2016).

Von Ahn, L. and L. Dabbish, "Labeling images with a computer game", in "Proc. the SIGCHI conf. Human factors in computing systems", pp. 319–326 (2004).

Wang, J. and G. Joshi, "Adaptive communication strategies to achieve the best error-runtime trade-off in local-update sgd", arXiv preprint arXiv:1810.08313 (2018a).

Wang, J. and G. Joshi, "Cooperative sgd: A unified framework for the design and analysis of communication-efficient sgd algorithms", arXiv preprint arXiv:1808.07576 (2018b).

Wang, W., L. Ying and J. Zhang, "A game-theoretic approach to quality control for collecting privacy-preserving data", in "Proc. Annu. Allerton Conf. Communication, Control and Computing", (Monticello, IL, 2015).

Wang, W., L. Ying and J. Zhang, "The value of privacy: Strategic data subjects, incentive mechanisms and fundamental limits", in "Proc. Ann. ACM SIGMETRICS Conf.", (Antibes Juan-les-Pins, France, 2016).

Wauthier, F. L. and M. I. Jordan, "Bayesian bias mitigation for crowdsourcing", in "Advances Neural Information Processing Systems (NIPS)", pp. 1800–1808 (2011).

Witkowski, J. and D. C. Parkes, "A robust bayesian truth serum for small populations.", in "AAAI Conf. Artificial Intelligence", vol. 12, pp. 1492–1498 (2012).

Yang, Q., Y. Liu, T. Chen and Y. Tong, "Federated machine learning: Concept and applications", ACM Transactions on Intelligent Systems and Technology (TIST) **10**, 2, 12 (2019).