A Bandit Framework for Strategic Regression

Yang Liu and Yiling Chen
School of Engineering and Applied Science, Harvard University
{yangl,yiling}@seas.harvard.edu

Abstract

We consider a learner's problem of acquiring data dynamically for training a regression model, where the training data are collected from strategic data sources. A fundamental challenge is to incentivize data holders to exert effort to improve the quality of their reported data, despite that the quality is not directly verifiable by the learner. In this work, we study a dynamic data acquisition process where data holders can contribute multiple times. Using a bandit framework, we leverage on the long-term incentive of future job opportunities to incentivize high-quality contributions. We propose a Strategic Regression-Upper Confidence Bound (SR-UCB) framework, an UCB-style index combined with a simple payment rule, where the index of a worker approximates the quality of his past contributions and is used by the learner to determine whether the worker receives future work. For linear regression and certain family of non-linear regression problems, we show that SR-UCB enables a $\mathrm{O}\big(\sqrt{\log T/T}\big)$-Bayesian Nash Equilibrium (BNE) where each worker exerting a target effort level that the learner has chosen, with T being the number of data acquisition stages. The SR-UCB framework also has some other desirable properties: (1) The indexes can be updated in an online fashion (hence computationally light). (2) A slight variant, namely Private SR-UCB (PSR-UCB), is able to preserve $\big(\mathrm{O}\big(\log^{-1} T\big), \mathrm{O}\big(\log^{-1} T\big)\big)$-differential privacy for workers' data, with only a small compromise on incentives (achieving $\mathrm{O}\big(\log^6 T/\sqrt{T}\big)$-BNE).

1 Introduction

More and more data for machine learning nowadays are acquired from distributed, unmonitored and strategic data sources and the quality of these collected data is often unverifiable. For example, in a crowdsourcing market, a data requester can pay crowd workers to label samples. While this approach has been widely adopted, crowdsourced labels have been shown to degrade the learning performance significantly, see e.g., [19], due to the low quality of the data. How to incentivize workers to contribute high-quality data is hence a fundamental question that is crucial to the long-term viability of this approach.

Recent work [2, 4, 10] has considered incentivizing data contributions for the purpose of estimating a regression model. For example Cai et al. [2] design payment rules so that workers are incentivized to exert effort to improve the quality of their contributed data, while Cummings et al. [4] design mechanisms to compensate privacy-sensitive workers for their privacy loss when contributing their data. These studies focus on a static data acquisition process, only considering one-time data acquisition from each worker. Hence, the incentives completely rely on the payment rule. However, in stable crowdsourcing markets, workers return to receive additional works. Future job opportunities are thus another dimension of incentives that can be leveraged on to motive high-quality data contributions. In this paper, we study dynamic data acquisition from strategic agents for regression problems and explore the use of future job opportunities to incentivize effort exertion.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

In our setting, a learner has access to a pool of workers and in each round decides on which workers to ask for data. We propose a Multi-armed Bandit (MAB) framework, called Strategic Regression-Upper Confidence Bound (SR-UCB), that combines a UCB-style index rule with a simple per-round payment rule to align the incentives of data acquisition with the learning objective. Intuitively, each worker is an arm and has an index associated with him that measures the quality of his past contributions. The indexes are used by the learner to select workers in the next round. While MAB framework is natural for modeling selection problem with data contributors of potentially varying qualities, our setting has two challenges that are distinct from classical bandit settings. First, after a worker contributes his data, there is no ground-truth observation to evaluate how well the worker performs (or reward as commonly referred to in a MAB setting). Second, a worker's performance is a result of his strategic decision, e.g., how much effort he exerts, instead of being purely exogenously. Our SR-UCB framework overcomes the first challenge by evaluating the quality of an agent's contributed data against an estimator trained on data provided by all other agents to obtain an unbiased estimate of the quality, an idea inspired by the peer prediction literature [11, 16]. To address the second challenge, our SR-UCB framework enables a game-theoretic equilibrium with workers exerting target effort levels chosen by the learner. More specifically, in addition to proposing the SR-UCB framework, our contributions include:

- We show SR-UCB helps simplify the design of payment, and successfully incentivizes effort exertion for acquiring data for linear regression. Every worker exerting a targeted effort level (for labeling and reporting the data) is a $\mathrm{O}\big(\sqrt{\log T/T}\big)$-Bayesian Nash Equilibrium (BNE). We can also extend the above results to a certain family of non-linear regression problems.
- SR-UCB indexes can be maintained in an online fashion, hence is computationally light.
- We extend SR-UCB index policy to further provide privacy guarantees (PSR-UCB), without hurting the provided incentive much. PSR-UCB is $(\mathrm{O}\big(\log^{-1} T\big), \mathrm{O}\big(\log^{-1} T\big))$-differentially private and every worker exerting the targeted effort level is a $\mathrm{O}\big(\log^6 T/\sqrt{T}\big)$-BNE.

2 Related work

Recent works have formulated various strategic learning settings under different objectives [2, 4, 10, 21]. Among these, payment based solutions are proposed for regression problems when data come from workers who are either effort sensitive [2] or privacy sensitive [4]. These solutions achieve game-theoretic equilibria for guaranteeing the quality of the contributed data. The basic idea is inspired by a much older and mature research literature, namely proper scoring rules [8] and peer prediction [16]. Both works consider a static data acquisition procedure, while our work focuses on a dynamic data acquisition process for regression problems. By leveraging on the long-term incentive of future job opportunities, our work has a much simpler payment rule than those of [2] and [4] and relaxes some of the restrictions on the learning objectives (e.g., well behaved [2]), at the cost of a weaker equilibrium concept (approximate BNE in this work vs. dominate-strategy in [2]).

MAB is a sequential decision making and learning framework which has been extensively studied. It is nearly impossible to survey the entire bandit literature, but it starts roughly with the seminal work by Lai et al [13], where lower and upper bounds on asymptotic regret on bandit selection are derived. More recently, finite time algorithms have been developed in [1] for i.i.d. bandits. Different from the classical settings, in this work we need to deal with challenges such as no ground-truth observations for bandits, as well as bandit's rewards being strategically decided. A couple of recent works [7, 15] also considered bandit setting with strategic arms. Our work differs from them in that we consider a regression learning setting without ground-truth observations, and also we consider long-term workers whose decisions on reporting data can change over time.

Our work and motivations have some resemblance to online contract design problems for a principal-agent model [9]. But unlike the online contract design problems, our learner cannot verify the quality of finished work after each task assignment. Also instead of focusing on learning the optimal contract, we use bandit to mainly maintain a long-term incentive to induce high-quality data.

3 Formulation

The learner observes a set of features data X for training. To make our analysis tractable, we assume each $x \in X$ is sampled uniformly from a unit ball with dimension d: $x \in \mathbb{R}^d$ s.t. $||x||_2 \leq 1$. Each

x associates with a ground-truth response (or label) $y(x)$, which cannot be observed directly by the learner. Suppose x and $y(x)$ are related through a function $f : \mathbb{R}^d \to \mathbb{R}$ that $y(x) = f(x) + z$, where z is an i.i.d. zero-mean noise with variance σ_z. For example, for linear regression $f(x) = \theta^T x$ for some $\theta \in \mathbb{R}^d$. The learner would like to learn a good estimate $\tilde{f}$ of f. In order to do so, the learner needs to figure out $y(x)$ for different $x \in X$ for training purpose. To obtain an estimate $\tilde{y}(x)$ of $y(x)$, the learner assigns each x to a selected worker to obtain a label.

Agent model: Suppose we have a set of workers $\mathcal{U} = \{1, 2, ..., N\}$ with $N \geq 2$. After receiving the task, each worker will decide on the effort level e he wants to exert to generate an outcome – higher effort leads to a better outcome, but also incurs higher cost. We assume e has bounded support $[0, \bar{e}]$ for all worker $i \in \mathcal{U}$. Each worker's decision on effort exertion is affected by his incentives in the market. In this paper we consider the model that each worker wants to maximize his expected payment minus cost for effort exertion. The labeling outcome $\tilde{y}(x)$ will be given back to the learner. Denote by $\tilde{y}_i(x, e)$ the label returned by worker i for data instance x (if assigned) with chosen effort level e. We consider the following effort-sensitive agent model: $\tilde{y}_i(x, e) = f(x) + z + z_i(e)$, where $z_i(e)$ is a zero-mean noise with variance $\sigma_i(e)$. Note $\sigma_i(e)$ can be different for different workers, and $\sigma_i(e)$ is decreasing in $e, \forall i$. All z, z_is have bounded support such that $|z|, |z_i| \leq Z$. We will be assuming the cost for exerting e amount of effort is simply e for every worker.

Learner's objective Suppose the learner wants to learn f with $|X|$ samples. Then the learner finds effort levels $\mathbf{e}^*$ for each data point such that

$$\mathbf{e}^* \in \text{argmin}_{\{e(x)\}_{x \in X}} \texttt{ERROR}(\tilde{f}(\{x, \tilde{y}(x, e(x))\}_{x \in X})) + \lambda \cdot \texttt{PAYMENT}(\{e(x)\}_{x \in X}) \ ,$$

where $e(x)$ is the effort level for sample x, and $\{\tilde{y}(x, e(x))\}_{x \in X}$ is the set of labeled responses for training data X. $\tilde{f}(\cdot)$ is the regression model trained over this data. The learner assigns the data and pay appropriately to induce the corresponding effort level $\mathbf{e}^*$. This formulation is not unlike the one presented in [2]. The `ERROR` term captures the expected error of the trained model using collected data (e.g., measure in squared loss), while the `PAYMENT` term captures the total expected budget learner spends to receive the data. This payment quantity depends on the mechanism that the learner chooses to use and is the expected payment of the mechanism to induce selected effort level for each data point $\{e(x)\}_{x \in X}$. $\lambda > 0$ is a weighting factor, which is a constant. It is clear that the objective function depends on σ_is. We assume for now that the learner knows $\sigma_i(\cdot)$s[1], and the optimal $\mathbf{e}^*$ can be computed.

4 StrategicRegression-UCB (SR-UCB): A general template

We propose SR-UCB for solving the dynamic data acquisition problem. SR-UCB enjoys a bandit setting, where we borrow the idea from classical UCB algorithm [1], which maintains an index for each arm balancing exploration and exploitation. While a bandit framework is not necessarily the best solution for our dynamic data acquisition problem, we provide reasoning on why a bandit framework can serve as a promising option. As utility maximizers, workers would like to be assigned tasks, if the marginal gain for taking a task in positive. A bandit algorithm can help execute the assignment process. Meanwhile the arm selection (of bandit algorithms) thus introduces competition among workers in improving their indexes, which the selection is based upon. When such indexes are well designed, the competition will be reflecting on the amount of efforts exerted by agents.

SR-UCB contains the following two critical components:

Per-round payment For each worker i, once selected to label a sample x, we will assign a base payment $p_i = e_i + \gamma$, [2] after reporting the labeling outcome, where e_i is the desired effort level that we would like to induce from worker i (for simplicity we have assumed the cost for exerting effort e_i equals to the effort level), and $\gamma > 0$ is a small quantity. The design of this base payment is to ensure once selected, a worker's base cost will be covered. Note the above payment depends on neither the assigned data instance x nor the reported outcome $\tilde{y}$. Therefore such a payment procedure can be pre-defined after the learner sets a target effort level.

[1] This assumption can be relaxed. See our supplementary materials for the case with homogeneous σ.

[2] We assume workers have knowledge of how the mechanism sets up this γ.

Assignment The learner assigns multiple task $\{x_i(t)\}_{i\in d(t)}$ at time t, with $d(t)$ denoting the set of workers selected at t. Denote by $e_i(t)$ the effort level worker i exerted for $x_i(t)$, if $i \in d(t)$. Note all $\{x_i(t)\}_{i\in d(t)}$ are different tasks, and each of them is assigned to exactly one worker. The selection of workers will depend on the notion of indexes. Details are given in Algorithm 1.

Algorithm 1 SR-UCB: Worker index & selection

Step 1. For each worker i, first train estimator $\tilde{f}_{-i,t}$ using data $\{x_j(n) : 1 \le n \le t-1,\ j \in d(n), j \neq i\}$, that is using the data collected from workers $j \neq i$ up to time t. When $t = 1$, we will initialize by sampling each worker at least once such that $\tilde{f}_{-i,t}$ can be computed.
Step 2. Then compute the following index for worker i at time t

$$I_i(t) = \frac{1}{n_i(t)} \sum_{n=1}^{t} 1(i \in d(n)) \left[a - b\left(\tilde{f}_{-i,t}(x_i(n)) - \tilde{y}_i(n, e_i(n)) \right)^2 \right] + c\sqrt{\frac{\log t}{n_i(t)}},$$

where $n_i(t)$ is the number of times worker i has been selected up to time t. a, b are two positive constants for "scoring", and c is a normalization constant. $\tilde{y}_i(n, e_i(n))$ is the corresponding label for task $x_i(n)$ with effort level $e_i(n)$, if $i \in d(n)$.
Step 3. Based on above index, we will select $d(t)$ at time t such that $d(t) := \{j : I_j(t) \ge \max_i I_i(t) - \tau(t)\}$, where $\tau(t)$ is a perturbation term decreasing in t.

Remarks are in order. (1) Different from classical bandit setting, when calculating the indexes, there is no ground-truth observation we can make to evaluate the performance of each worker. Therefore we adopt the notion of scoring rule [8]. Particularly the one we used above is the well-known Brier scoring rule: $B(p,q) = a - b(p-q)^2$. (2) The scoring rule based index looks similar to the payment strategy studied in [2,4]. But as we will show later, under our framework the selection of a, b is much less sensitive to different problem settings, as with an index policy, only the relative values matter (ranking). This is another benefits of separating payment from selection. (3) We are not going to only select the best arm with the highest index. Instead we are going to select workers whose index is within a certain range of the maximum one (a confidence region) – workers may have competing expertise level, selecting only one of them will de-incentivize workers' effort exertion.

4.1 Solution concept

Denote by $\mathbf{e}(n) := \{e_1(n), ..., e_N(n)\}$, and $e_{-i}(n) = \{e_j n\}_{j\neq i}$. We define approximate Bayesian Nash Equilibrium as our solution concept:

Definition 1. Suppose SR-UCB runs for T stages. $\{e_i(t)\}_{i=1,t=1}^{N,T}$ is a π-BNE if $\forall i, \{\tilde{e}_i(t)\}_{t=1}^{T}$:

$$\frac{1}{T}\mathbb{E}[\sum_{t=1}^{T}(p_i - e_i(t))1(i \in d(t))\big|\{\mathbf{e}(n)\}_{n\le t}] \ge \frac{1}{T}\mathbb{E}[\sum_{t=1}^{T}(p_i - \tilde{e}_i(t))1(i \in d(t))\big|\{\tilde{e}_i(n), e_{-i}(n)\}_{n\le t}] - \pi.$$

This is to say by deviating, each worker will gain π more net-payment per around. We will establish our main results in terms of π-BNE. The reason we adopt such a notion is in a sequential setting it is generally hard to achieve strict BNE or even other stronger notion, as any one step deviation may not affect a long term evaluation by much. [3] In this regard, approximate BNE is arguably the best solution concept we can hope for.

5 Linear regression

5.1 Settings and a warm up scenario

In this section we start presenting our results for a simple linear regression task such that the feature x and observation y are linearly related via an unknown θ: $y(x) = \theta^T x + z,\ \forall x \in X$. Let's start with assuming all workers are statistically identical such that $\sigma_1 = \sigma_2 = ... = \sigma_N$. This is an easier case to start with to serve as a warm up. It is known that given training data, we can find an estimation $\tilde{\theta}$

[3] Certainly we can re-run any BNE or dominant strategy for a one shot setting, e.g. [2], for every time step. But such solution does not incorporate long term incentives.

that minimizes a non-regularized empirical risk function: $\tilde{\theta} = \text{argmin}_{\hat{\theta}\in\mathbb{R}^d} \sum_{x\in X}(y(x) - \hat{\theta}^T x)^2$ (linear least square). To put this model into SR-UCB, denote $\tilde{\theta}_{-i}(t)$ as the linear least square estimator trained using data from workers $j \neq i$ up to time $t-1$. And $I_i(t) := S_i(t) + c\sqrt{\log t / n_i(t)}$, with

$$S_i(t) := \frac{1}{n_i(t)} \sum_{n=1}^{t-1} 1(i \in d(n)) \left[a - b\left(\tilde{\theta}_{-i}^T(t) x_i(n) - \tilde{y}_i(n, e_i(n)) \right)^2 \right]. \tag{5.1}$$

Since $||x||_2 \leq 1$ and suppose $||\theta||_2 \leq M$, we then have $\forall t, n, i$, $(\tilde{\theta}_{-i}^T(t) x_i(n) - \tilde{y}_i(n, e_i(n)))^2 \leq 8M^2 + 2Z^2$. Choose a, b such that $a - (8M^2 + 2Z^2)b \geq 0$, then we have $0 \leq a - (8M^2 + 2Z^2)b \leq S_i(t) \leq a$, $\forall i, t$. Set $\tau(t) := \mathrm{O}\big(\sqrt{\log t / t}\big)$ – the basic idea is that with t number of samples, the uncertainties in the indexes can be upper bounded at the order of $\mathrm{O}\big(\sqrt{\log t / t}\big)$, including both the uncertainties coming from score calculation and bias term. Thus, to not miss a competitive worker, we set the tolerance to be at the same order. We first make the following assumption on the smoothness of σ.

Assumption 1. *We assume $\sigma(e)$ is convex on $e \in [0, \bar{e}]$, with gradient $\sigma'(e)$ being both upper bounded, and lower bounded away from 0, i.e., $\bar{L} \geq |\sigma'(e)| \geq \underline{L} > 0$, $\forall e$.*

The learner wants to learn f with total NT ($= |X|$ or $\lceil NT \rceil = |X|$) samples, ideally T from each worker (since all workers are statistically equivalent). We run SR-UCB for T steps (and the learner knows T). Then the learner finds an effort level e^* such that

$$e^* \in \text{argmin}_e \mathbb{E}_{x,y,\tilde{y}} \left[\theta^T (\{x_i(n), \tilde{y}_i(n,e)\}_{i=1,n=1}^{N,T}) \cdot x - y \right]^2 + \lambda \cdot (e + \gamma) NT$$

Theorem 1. *Under SR-UCB for linear least square, set fixed payment $p_i, \forall i$ as follows: $p_i = e^* + \gamma$, $\gamma = \Omega(\sqrt{\log T / T})$, and set c to be large enough $c \geq$ Const.(M, Z, N, b), $\tau(t) := \mathrm{O}\big(\sqrt{\log t / t}\big)$. Workers have full knowledge of above. Then exerting effort $e_i(t) \equiv e^*, \forall t$ is $\mathrm{O}\big(\sqrt{\log T / T}\big)$-BNE $\forall i$.*

The marginal payment (payment minus the effort cost) per task can be made arbitrarily small by setting γ exactly on the order of $\mathrm{O}\big(\sqrt{\log T / T}\big)$, and $p_i - e^* = \gamma = \mathrm{O}\big(\sqrt{\log T / T}\big) \to 0$, as $T \to \infty$.

Our solution concept relies heavily on forming a race among workers. By establishing the convergence of bandit indexes to a function of effort (via $\sigma(\cdot)$), we show when the other workers $j \neq i$ follow the equilibria for exerting effort, worker i will be selected w.h.p. at each round, if he also puts in the same amount of effort. On the other hand, if worker i shirks from doing so by much ($\mathrm{O}\big(\sqrt{\log T / T}\big)$), his number of selection will go down in order. This establishes the π-BNE. Though as will be shown in next section, as long as there exists one competitive worker, all others will be incentivized to exert effort, it is still true that if all workers agree to shirk from exerting effort, they will arrive at a similar equilibria as we proved above (equally in-competitive is as good as equally competitive). This caveat can be removed with the following remedy. When there are ≥ 2 workers being selected as winners, each of them will be assigned tasks with certain probability $0 < p_s < 1$. While when there is a single winner, the winner will be selected w.p. 1. Set $p_s := 1 - \mathrm{O}\big(\sqrt{\log T / T} / \gamma\big)$. So with probability $1 - p_s = \mathrm{O}\big(\sqrt{\log T / T} / \gamma\big)$, even the "winning"workers will miss the selection. This process is independent of the bandit selection procedure. With above change, exerting e^* still leads to a $\mathrm{O}\big(\sqrt{\log T / T}\big)$-BNE, while (every worker) exerting any effort level that is $\Delta e > \mathrm{O}\big(\gamma\big)$ away from target effort level is not a π-BNE with $\pi \leq \mathrm{O}\big(\sqrt{\log T / T}\big)$.

Equilibria with heterogeneous effort level So far we only discussed about the equilibria at which workers exert consistent level of efforts over time. As what we can similarly show in the proof for Theorem 1, if every other worker $j \neq i$ is exerting the same level of effort over time, deviating to any other heterogeneous effort exertion strategy will not help worker i gain much in his average profit ($\mathrm{O}\big(\sqrt{\log T / T}\big)$). Analyzing the case when every worker is exerting different effort level at different time is challenging, so is on workers' side. This remains an interesting future direction.

5.2 Linear regression with different σ

Now we consider the more realistic case that different workers have different noise-effort function σs. W.l.o.g., we assume $\sigma_1(e) < \sigma_2(e) < ... < \sigma_N(e), \forall e$[4]. In such a setting, ideally we would always

[4]Combing with the results for homogeneous workers, we can again easily extend the results to the case with workers have both same and different expertise level.

like to collect data from worker 1 since he has the best expertise level (lowest variance in labeling noise). Suppose we are targeting an effort level e_1^* from data source 1 (the best data source). We first argue that we also need to incentivize worker 2 to exert competitive effort level e_2^* such that $\sigma_1(e_1^*) = \sigma_2(e_2^*)$, and we do assume such a e_2^* exists[5]. This also naturally implies that $e_2^* > e_1^*$ as worker 1 contributes data with less variance in noise at the same effort level. The reason is similar to the homogeneous setting – over time workers form a competition on $\sigma_i(e_i)$. Having a competitive peer will motivate workers to exert as much effort as he can (up to the payment). Therefore the goal for such a learner (with $2T$ samples to assign) is to find a effort level e^* such that [6]

$$e^* \in \text{argmin}_{e_2:\sigma_1(e_1)=\sigma_2(e_2)} \mathbb{E}_{x,y,\tilde{y}} \left[\theta^T (\{x_i(n), \tilde{y}_i(n,e_i))\}_{i=1,n=1}^{2,T}) x - y \right]^2 + \lambda \cdot (e_2 + \gamma) 2T.$$

Set one step payment to be $p_i = e^* + \gamma, \forall i$. Denote $\mathbf{e}^* = \{e_1(t) \equiv \{e_1^* : \sigma_1(e_1^*) = \sigma_2(e^*)\},\ e_i(t) \equiv e^*,\ \forall i \geq 2\}_t$. Note for $i > 2$ we have $\sigma_i(e^*) - \sigma_1(e_1^*) > 0$. While we have argued about the necessity for choosing the top two most competitive workers, we haven't mentioned the optimality of doing so. In fact selecting the top two is the best we can do. Suppose on the contrary, the optimal solution is by selecting top $k > 2$ workers, at effort level e_k. According to our solution, we targeted the effort level that leads to variance of noise $\sigma_k(e_k)$ (so the least competitive worker will be incentivized). Then we can simply target the same effort level e_k, but migrating the task loads to only the top two workers – this keeps the payment the same, but the variance of noise now becomes $\sigma_2(e_k) < \sigma_k(e_k)$, which leads to better performance. Denote $\Delta_1 := \sigma_3(e_1^*) - \sigma_1(e^*) > 0$ and assume Assumption 1 applies to all σ_is. We prove:

Theorem 2. *Under SR-UCB for linear least square, set $c \geq$ Const.(M, Z, b, Δ_1), $\Omega(\sqrt{\log T/T}) = \gamma \leq \frac{\Delta_1}{2L}$, $\tau(t) := \mathrm{O}\big(\sqrt{\log t/t}\big)$, exerting efforts following $\mathbf{e}^*$ is $\mathrm{O}\big(\sqrt{\log T/T}\big)$-BNE for all workers.*

Performance with acquired data If workers follow the π-BNE, the contributed data from the top two workers (who have been selected the most number of times) will have the same variance $\sigma_1(e_1^*)$. Then following results in [4], the performance of the trained classifier is bounded by $\mathrm{O}\big(\sigma_1(e_1^*)/(\sum_{i=1,2} n_i(T))^2\big)$ w.h.p. Ideally we want to have $\sum_{i=1,2} n_i(T) = 2T$. The expected performance loss (due to missed sampling & wrong selection, which is bounded at the order of $\mathrm{O}\big(\log T\big)$) is bounded by $\mathbb{E}[\sigma_1(e_1^*)/(\sum_{i=1,2} n_i(T))^2 - \sigma_1(e_1^*)/(2T)^2] \leq \mathrm{O}\big(\sigma(e^*) 2 \log T/T^3\big)$ w.h.p. .

Regularized linear regression Ridge estimator has been widely adopted for solving linear regression. The objective is to find a linear model $\tilde{\theta}$ that minimizes the following regularized empirical risk: $\tilde{\theta} = \text{argmin}_{\hat{\theta} \in \mathbb{R}^d} \sum_{x \in X} (y(x) - \hat{\theta}^T x)^2 + \rho ||\hat{\theta}||_2^2$, with $\rho > 0$ being the regulation parameter. We claim that simply changing the $\tilde{f}_{-i,t}(\cdot)$ in SR-UCB to the output from above ridge regression, the $\mathrm{O}\big(\sqrt{\log T/T}\big)$-BNE for inducing an effort level e^* will hold. Different from the non-regularized case, the introduce of the regulation term will add bias in $\tilde{\theta}_{-i}^T(t)$, which gives a biased evaluation of indexes. Such bias poses additional challenge for payment function design in static data acquisition. We find proving the convergence of $\tilde{\theta}_{-i}^T(t)$ (so again the indexes will converge properly) enables an easy adaption of our previous results for non-regularized case to ridge regression:

Lemma 1. *With n i.i.d. samples w.p. $\geq 1 - e^{-Kn}$ ($K > 0$ is a constant), $||\tilde{\theta}_{-i}(t) - \theta||_2^2 \leq \mathrm{O}\big(\frac{1}{n^2}\big)$.*

Non-linear regression The basic idea for extending the results to non-linear regression is inspired by the consistency results on M-estimator [14], when the error of training data satisfies zero mean. Similar to the reasoning for Lemma 1, if $(\tilde{f}_{-i,t}(x) - f(x))^2 \to 0$, we can hope for an easy adaptation of our previous results. Suppose the non-linear regression model can be characterized by a parameter family Θ, where f is characterized by parameter θ, and $\tilde{f}_{-i,t}$ by $\tilde{\theta}_i(t)$. Due to the consistency of M-estimator we will be having $||\tilde{\theta}_i(t) - \theta||_2 \to 0$; More specifically, according to the results from [18], for non-linear regression model we can establish a $\mathrm{O}\big(1/\sqrt{t}\big)$ convergence rate with t training samples. When f is Lipschitz in parameter space, i.e., there exists a constant $L_N > 0$ such that $|\tilde{f}_{-i,t}(x) - f(x)| \leq L_N ||\tilde{\theta}_i(t) - \theta||_2$. By dominant convergence theorem we also have $(\tilde{f}_{-i,t}(x) - f(x))^2 \to 0$, and $(\tilde{f}_{-i,t}(x) - f(x))^2 \leq \mathrm{O}\big(1/t\big)$. The rest of the proof can then follow.

Example 1. Logistic function $f(x) = \frac{1}{1+e^{-\theta^T x}}$ satisfies Lipschitz condition with $L_N = 1/4$.

[5] When the supports for $\sigma_1(\cdot), \sigma_2(\cdot)$ overlap for a large support range.

[6] Since we only target the top two workers, we can limit the number of acquisitions on each stage to be no more than two, so the number of query does not go beyond $2T$.

6 Computational issues

The bandit framework provides us a nice way for building a "reputation system" for data market to incentivize effort using long term incentive, thus addressing one of the challenges brought up in [20]. Nevertheless, in order to update the indexes and select workers adaptively, we suffer from a couple of computational issues. First in order to update the index for each worker at any time t, a new estimator $\tilde{\theta}_{-i}(t)$ (using data from all other workers $j \neq i$ up to time $t-1$) needs to be re-computed. Secondly we need to re-apply $\tilde{\theta}_{-i}(t)$ to every collected sample from worker $i, \{(x_i(n), \tilde{y}_i(n, e_i(n)) : i \in d(n), n = 1, 2, ...t-1\}$ from previous rounds. We propose online variants of SR-UCB.

Online update of $\tilde{\theta}_{-i}(\cdot)$ Thanks to results from online learning literature, instead of re-computing $\tilde{\theta}_{-i}(t)$ at each step, which involves re-calculating the inverse of a covariance matrix (e.g., $(\rho I + X^T X)^{-1}$ for ridge regression) whenever there is a new sample point arriving, we can update $\tilde{\theta}_{-i}(t)$ in an online fashion, which is computationally much more efficient. We demonstrate our results with ridge linear regression. Start with an initial model $\tilde{\theta}^{\text{online}}_{-i}(1)$. Denote by $(x_{-i}(t), \tilde{y}_{-i}(t))$ any newly arrived sample at time t from worker $j \neq i$. Update $\tilde{\theta}^{\text{online}}_{-i}(t+1)$ (for computing $I_i(t+1)$) as [17]:

$$\tilde{\theta}^{\text{online}}_{-i}(t+1) := \tilde{\theta}^{\text{online}}_{-i}(t) - \eta_t \cdot \nabla_{\tilde{\theta}^{\text{online}}_{-i}(t)} \left[(\theta^T x_{-i}(t) - \tilde{y}_{-i}(t))^2 + \rho ||\theta||_2^2 \right],$$

Notice there could be multiple such data points arriving at each time – in which case we will update sequentially in an arbitrarily order. It is also possible that there is no sample point arriving from workers other than i at a time t, in which case we simply do not perform an update. Name this online updating SR-UCB as OSR1-UCB. With online updating, the accuracy of trained model $\tilde{\theta}^{\text{online}}_{-i}(t+1)$ converges slower, so is the accuracy in the indexes for characterizing workers' performance. Nevertheless we prove exerting targeted effort exertion e^* is $\mathrm{O}\big(\sqrt{\log T/T}\big)$-BNE under OSR1-UCB for ridge regression, using convergence results for $\tilde{\theta}^{\text{online}}_{-i}(t)$ proved in [17].

Online score update Online updating can also help compute $S_i(t)$ (in $I_i(t)$) efficiently. Instead of repeatedly re-calculating the score for each data point (in $S_i(t)$), we only update the newly assigned samples which has not been evaluated yet, by replacing $\tilde{\theta}^{\text{online}}_{-i}(t)$ with $\tilde{\theta}^{\text{online}}_{-i}(n)$ in $S_i(t)$:

$$S^{\text{online}}_i(t) := \frac{1}{n_i(t)} \sum_{n=1}^{t-1} 1(i \in d(n)) \left[a - b \left((\tilde{\theta}^{\text{online}}_{-i}(n))^T x_i(n) - \tilde{y}_i(n, e_i(n)) \right)^2 \right]. \tag{6.1}$$

With less aggressive update, again the index terms' accuracies converge slower than before, which is due to the fact the older data is scored using an older (less accurate) version of $\tilde{\theta}^{\text{online}}_{-i}$ without being further updated. We propose OSR2-UCB where we change the index and bias term of SR-UCB to: $S^{\text{online}}_i(t) + c\sqrt{(\log t)^2/n_i(t)}$, to deal with the slower convergence of indexes. We establish $\mathrm{O}\big(\log T/\sqrt{T}\big)$-BNE for worker's effort exertion –the change is due to the change of the bias term.

7 Privacy preserving SR-UCB

With a repeated data acquisition setting, workers' privacy in data may leak repeatedly. In this section we study an extension of SR-UCB to preserve privacy of each individual worker's contributed data. Denote the training data collected as $\mathcal{D} := \{\tilde{y}_i(t, e_i(t))\}_{i \in d(t), t}$. We quantify privacy using differential privacy [5], and we adopt (ε, δ)-differential privacy (DP) [6], which is defined as follows:

Definition 2. A mechanism $\mathcal{M} : \mathcal{D} \to \mathbb{R}$ is (ε, δ)-differentially private if for any $i \in d(t), t$, any two distinct $\tilde{y}_i(t, e_i(t)), \tilde{y}'_i(t, e'_i(t))$, and for every subset of possible outputs $\mathcal{S} \subseteq \mathbb{R}$, $\Pr[\mathcal{M}(D) \in S] \leq \exp(\varepsilon) \Pr[\mathcal{M}(D \backslash \{\tilde{y}_i(t, e_i(t))\}, \tilde{y}'_i(t, e'_i(t))) \in S] + \delta$.

Suppose the learner will protect workers' privacy (e.g., companies will keep customers' data in private), so contributing data to the leaner directly will not leak a particular worker's privacy. The privacy leakage occurs in two ways: (1) The learned regression model $\tilde{\theta}(T)$, which is trained using all data collected after T rounds. Suppose after learning the regression model $\tilde{\theta}(T)$, this information will be released for public usage or monitoring. This information contains each individual worker's private information. Note this is a one shot leak of privacy (published at the end of the training (step T)). (2) The second ones are the indexes. Each worker i's data will be utilized towards calculating other workers' indexes $I_j(t), j \neq i$, as well as his own $I_i(t)$, which will be published.[7] Note this type

[7] It is debatable whether the indexes should be published or not. But revealing decisions on worker selection will also reveal information on the indexes. We consider the more direct scenario, where they are published.

of leakage occurs at each step. To simplify the matter, instead of $I_j(t)$, we can focus on the privacy losses in $S_j(t)$, as $I_j(t)$ is a function of $S_j(t)$ and $n_i(t)$, and we prove the following:

Lemma 2. *At any time t, $\forall i$, $n_i(t)$ can be written as a function of $\{S_j(n), n < t\}_j$.*

Preserving privacy in $\tilde{\theta}(T)$ To protect privacy in $\tilde{\theta}(T)$, following standard method, we add a Laplacian noise vector $\mathbf{v}_\theta$ to it [6]: $\tilde{\theta}^p(T) = \tilde{\theta}(T) + \mathbf{v}_\theta$, where $\Pr(\mathbf{v}_\theta) \propto \exp(-\varepsilon_\theta ||\mathbf{v}_\theta||_2)$. $\varepsilon_\theta > 0$ is a parameter controlling the noise level.

Lemma 3. *Set $\varepsilon_\theta = 2\sqrt{T}$, the output $\tilde{\theta}^p(T)$ of SR-UCB for linear regression preserves $(\mathrm{O}(T^{-1/2}), exp(-\mathrm{O}(T)))$-DP. Further w.p. $\geq 1 - 1/T^2$, $||\tilde{\theta}^p(T) - \tilde{\theta}(T)||_2 = ||\mathbf{v}_\theta||_2 \leq \log T/\sqrt{T}$.*

Preserving privacy in $\{I_i(t)\}_{i,t}$: a continual privacy preserving model For indexes $\{I_i(t)\}_i$ it is tempting to add $v_i(t)$ to each index (and for selection) $I_i(t) := I_i(t) + v_i(t)$, where again $v_i(t)$ is a zero-mean Laplacian noise. However with releasing $\{I_i(t)\}_i$at each step, we will be releasing a noisy version of each $\tilde{y}_i(n, e_i(n)), i \in d(n), \forall n < t$. Then via composition theory in differential privacy [12], we know the preserved privacy level will grow in time t, unless we add significant noise on each stage – but this will completely destroy the informativeness of our index policy.

We borrow the partial sum idea from differential privacy results on continual observations [3]. The idea is when releasing continual data, instead of exerting noise at any step, the current to-be-released data will be decoupled into sum of partial sums, and we only add noise to each partial sums – hopefully the noisy version of the partial sums can be reused repeatedly. In order to implement this idea, first we need to transform the information to be published into partial sums. For each worker i and $S_i(t)$, if we adopt online update in Eqn. (6.1), it is fairly clear $S_i^{\text{online}}(t)$ can be written down as sum of partial sums of terms invoking $\tilde{y}_i(n, e_i(n))$. The basic idea works as follows: Write $S_i^{\text{online}}(t)$ as the a summation: $\sum_{n=1}^{t-1} dS(n)/n_i(t)$. Write down t as a binary string and flip its rightmost bit to 0: this gives $q(t)$. Take the sum from $q(t)+1$ to t: $\sum_{n=q(t)+1}^{t} dS(n)$ as one partial sum. Repeat above for $q(t)$, to get $q(q(t))$, and the second partial sum $\sum_{n=q(q(t))+1}^{q(t)} dS(n)$, until we reach $q(\cdot) = 0$. So

$$S_i^{\text{online}}(t) = \frac{1}{n_i(t)} \left(\sum_{n=q(t)+1}^{t} dS(n) + \sum_{n=q(q(t))+1}^{q(t)} dS(n) + \ldots + \sum_{n=0}^{0} dS(n) \right).$$

For each partial sum above we add a Laplacian noise v_S with distribution $\Pr(v_S) \propto e^{-\varepsilon |v_S|}$. With this we can hope to bound the number of total noise terms ($\leq \lceil \log t \rceil$ with t steps), as well as the number of appearance of each private data in the partial sums ($\leq \lceil \log t \rceil$ with t terms [3]). It is however not as clear as how to make such a decouple (into partial sums) for $\{\tilde{\theta}_{-j}^{\text{online}}(n)\}_{n=1}^t$ (in $S_j^{\text{online}}(t)$), which contains information of $\tilde{y}_i(n, e_i(n))$. In our solution, change $\tilde{\theta}_{-j}^{\text{online}}(t)$ to $\tilde{\theta}_{-j}^{\text{online}}(t) := \sum_{n=1}^t \tilde{\theta}_{-j}(n)/t$, where $\tilde{\theta}_{-j}(n)$ is the regression model we estimated using all data from worker $j \neq i$ up to time n. With this we apply the partial sum idea to $\sum_{n=1}^t \tilde{\theta}_{-j}(n)$ (add $\Pr(\mathbf{v}_{\tilde{\theta}}) \propto e^{-\varepsilon ||\mathbf{v}_{\tilde{\theta}}||_2}$ to each partial sum).

The rest we need to show is with above two noise exertion procedures, our index policy SR-UCB will not lose its value in incentivizing. We show in order to prove similar convergence results, we need to update SR-UCB by changing the index into the following format:

$$I_i(t) = \hat{S}_i^{\text{online}}(t) + c(\log^3 t \log^3 T)/\sqrt{n_i(t)},\ \tau(t) = \mathrm{O}\big((\log^3 t \log^3 T)/\sqrt{t}\big),$$

where $\hat{S}_i^{\text{online}}(t)$ denotes the noisy version of $S_i^{\text{online}}(t)$ with added noises ($v_S, \mathbf{v}_{\tilde{\theta}}$ etc). The change of bias is mainly to incorporate the increased uncertainty level (due to added privacy preserving noise). Denote this mechanism as PSR-UCB, we have:

Theorem 3. *Set $\varepsilon := 1/\log^3 T$ for added noises (both $v_S, \mathbf{v}_{\tilde{\theta}}$), PSR-UCB preserves $(\mathrm{O}(\log^{-1} T), \mathrm{O}(\log^{-1} T))$-DP for linear regression.*

With homogeneous workers, we similarly can prove exerting effort e^* (optimal effort level) is $\mathrm{O}(\log^6 T/\sqrt{T})$-BNE. We do see in order to protect privacy in the bandit setting, we loss in the approximation term of BNE (less incentive to provide).

Acknowledgement: We acknowledge the support of NSF grant CCF-1301976.

References

[1] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[2] Yang Cai, Constantinos Daskalakis, and Christos H Papadimitriou. Optimum statistical estimation with strategic data sources. *arXiv preprint arXiv:1408.2539*, 2014.

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

[4] Rachel Cummings, Stratis Ioannidis, and Katrina Ligett. Truthful linear regression. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 448–483, 2015.

[5] Cynthia Dwork. Differential privacy. In *Automata, languages and programming*, pages 1–12. Springer, 2006.

[6] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy.

[7] Arpita Ghosh and Patrick Hummel. Learning and incentives in user-generated content: Multi-armed bandits with endogenous arms. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 233–246. ACM, 2013.

[8] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[9] Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. In *Proceedings of the fifteenth ACM EC*, pages 359–376. ACM, 2014.

[10] Stratis Ioannidis and Patrick Loiseau. Linear regression as a non-cooperative game. In *Web and Internet Economics*, pages 277–290. Springer, 2013.

[11] Radu Jurca and Boi Faltings. Collusion-resistant, incentive-compatible feedback payments. In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 200–209. ACM, 2007.

[12] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. *arXiv preprint arXiv:1311.0776*, 2013.

[13] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[14] Guy Lebanon. m-estimators and z-estimators.

[15] Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 565–582. ACM, 2015.

[16] Nolan Miller, Paul Resnick, and Richard Zeckhauser. Eliciting informative feedback: The peer-prediction method. *Management Science*, 51(9):1359–1373, 2005.

[17] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. *arXiv preprint arXiv:1109.5647*, 2011.

[18] BLS Prakasa Rao. The rate of convergence of the least squares estimator in a non-linear regression model with dependent errors. *Journal of Multivariate Analysis*, 14(3):315–322, 1984.

[19] Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM, 2008.

[20] Aleksandrs Slivkins and Jennifer Wortman Vaughan. Online decision making in crowdsourcing markets: Theoretical challenges. *ACM SIGecom Exchanges*, 12(2):4–23, 2014.

[21] Panos Toulis, David C. Parkes, Elery Pfeffer, and James Zou. Incentive-Compatible Experimental Design. *Proceedings 16th ACM EC'15*, pages 285–302, 2015.