Incentivizing and Rewarding High-Quality Data via Influence Functions

Anonymous Authors[1]

Abstract

We consider a crowdsourcing data acquisition scenario, such as federated learning, where a *center* collects data points from a set of rational *agents*, with the aim of training a model. We show how a payment structure can be designed to incentivize the agents to provide high-quality data, and to provide this data as early as possible, based on a characterization of the influence that data points have on the empirical risk of the model. Our contributions can be summarized as follows: (a) we prove theoretically that this scheme ensures *truthful* data reporting as a game-theoretic equilibrium and further demonstrate its robustness against mixtures of truthful and *heuristic* data reports, (b) we design a procedure for a subset of models according to which the influence computation can be efficiently approximated and processed sequentially in batches over time, and (c) we develop a theory for linear regression models that allows correcting the difference between the influence and the overall change in model risk.

1. Introduction

The success of machine learning depends to a large extent on the availability of high quality data. For many applications, data has to be elicited from independent and sometimes self-interested data providers. A good example is federated learning (Konečný et al., 2016), where a single *center* (e.g. a large company) collects data from a set of *agents* to jointly learn a model. Other examples of such settings can be found in *crowdsourcing*.

So far, research in federated learning has focused on protecting the privacy of contributed data, but has not considered how to reward the agents for the data they provided. Given that gathering accurate data is often a costly task, one should not expect the agents to exert the required effort unless they are properly compensated. This phenomenon has been documented in practice (e.g., see (Vuurens, de Vries, and Eickhoff, 2011; Shah, Zhou, and Peres, 2015) and references therein) and is justified by the principles of agent rationality, an integral part of the field of *game theory*. Game theory tells us that we should be rewarding agents monetarily, with their payments being dependent on the quality of the data they provide. However, to do this, we first need to answer the following question: ”What constitutes high-quality data?”

Intuitively, a high-quality data point is one that improves the accuracy of the model. We should be looking to give higher rewards to the agents that provide useful data, compared to those agents that do not contribute to improving the estimation. Additionally, we would like to reward more those agents that provide their data as early as possible.

1.1. Our Approach

We consider a crowdsourcing scenario like the one described above, and we aim to design incentive schemes, i.e., mechanisms that reward the agents proportionally to the effect that they have on the accuracy of the model. A clear way of measuring the effect of individual points on the accuracy of a model is via the classical notion of *influence* (Cook and Weisberg, 1980). For a given data point, the influence quantifies how much the model's predictions would change if that point were not used in the training process. This allows us to quantify the effect that a single point has on the final outcome; we can simply remove the point, retrain, and compare the difference in the loss function. Based on the influence, we design schemes that reward agents proportionally to the decrease in the loss function due to their provided data points. The center then will only have to pay for data that improves the model's predictive capabilities and has the guarantee that the total cost of the data acquisition process will be bounded. Fig. 1 shows how an influence-based crowdsourcing elicitation framework might be structured.

We will show theoretically that the prescribed behavior, i.e., exerting the required effort to extract a sample from the underlying distribution and reporting that sample, is the best option for the agents under reasonable assumptions. In game-theoretic terms, we prove that our incentive schemes induce this type of behavior as an equilibrium of the corre-

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

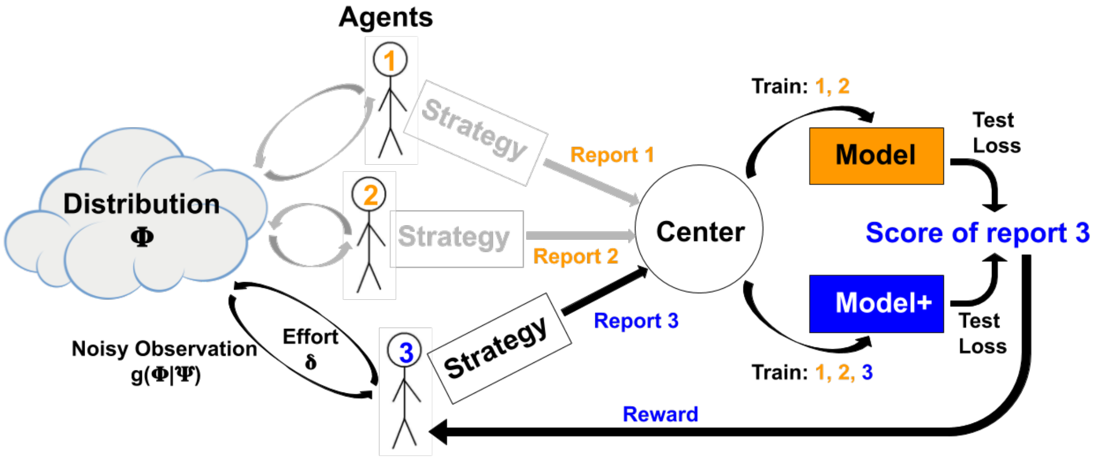

Figure 1: The setting in this paper: self-interested strategic agents observe a distribution and report to a center. The center learns a model and rewards agents (here agent 3) according the their quality score, given the other reports that were received earlier (here from agents 1 and 2).

sponding game. We strengthen our result by showing the robustness of the scheme against heuristic reports, i.e., agents that do not exert the effort to obtain useful information, and simply provide some untruthful report. Specifically, we show that (a) if the Center has independent data for testing, then the agents are always incentivized to provide truthful reports and (b) the same holds even if the test set is assembled by the reports of the agents, and a fraction of the agents decide to use an uninformed strategy.

For many practical applications computing the exact influence is prohibitively inefficient. For this reason, following Koh and Liang [2017], we compute an *approximation of the exact influence*, based on up-weighing the training point by a small quantity. Our proposed approximation extends the idea in (Koh and Liang, 2017), which (without any modifications) turns out to be insufficient for our purposes. We show that the employed approximation is very close to the value of the exact influence, while achieving a dramatic improvement in speed for gradient descent-based learning models such as logistic regression.

Then, we consider the case where the data points arrive sequentially; this captures most real-life scenarios of interest, as the data acquisition process is usually sequential over time. We employ our incentive scheme to reward the agents *in batches*, with the fundamental property that agents that provide their data earlier are rewarded more, a desirable property as explained above. We consider two alternatives when it comes to the influence of data points in the current batch: *Marginal-Loss* (M-Loss), in which we include the data points of the current batch in training and *Marginal-Gain* (M-Gain), in which we do not. We propose an analytical method, which we demonstrate for linear regression, that the Center can use to apply a correction factor to either M-Loss or M-Gain which guarantees a bounded budget. We run experiments on several different real datasets, as well as generated data, and verify the efficacy of this correction factor.

1.2. Related Work

The topic of learning a model when the input data points are provided by strategic sources has been the focus of a growing literature at the intersection of machine learning and game theory. A significant amount of work has been devoted to the setting in which agents are interested in the outcome of the estimation process itself, e.g., when they are trying to sway the learned model closer to their own data points (Perote and Perote-Pena, 2004; Dekel, Fischer, and Procaccia, 2010; Meir, Procaccia, and Rosenschein, 2012; Caragiannis, Procaccia, and Shah, 2016; Chen et al., 2018b). Our setting is concerned with the fundamental question of eliciting accurate data when data acquisition is costly for the agents, or when they are not willing to share their data without some form of monetary compensation. Another line of work considers settings in which the agents have to be compensated for their loss of privacy (Cummings, Ioannidis, and Ligett, 2015; Chen et al., 2018a).

A similar question to the one in our paper was considered by Cai, Daskalakis, and Papadimitriou [2015], where the authors design strategy-proof mechanisms for eliciting data and achieving a desired trade-off between the accuracy of the model and the payments issued. A similar model to (Cai, Daskalakis, and Papadimitriou, 2015) was considered in (Westenbroek et al., 2017; 2019), for the case of multiple Centers eliciting data from the crowd. Because of the close comparison, we will elaborate on this further in section 2.

Our ideas are closely related to the literature of *Peer Consistency* mechanisms (Faltings and Radanovic, 2017) and *Peer Prediction mechanisms for crowdsourcing* (Radanovic, Faltings, and Jurca, 2016). The idea behind this literature is to extract high-quality information from individuals by comparing their reports against those of randomly chosen peers. This approach has been largely successful in the theory of eliciting *truthful* information. The same principle applies to our case, where the payments are dependent on the improvement of the model and therefore agents are rewarded for providing *helpful* information. Finally, Jia et al. [2019] recently considered a setting in which the value of the provided data information is determined via the *Shapley value*. Their approach is inherently different from ours, but it is worth noting that they consider the influence approximation of (Koh and Liang, 2017) for approximating the Shapley value.

2. Setting

In our setting, there is a *Center* that wants to learn a model parametrized by θ, with a non-negative loss function $L(z, \theta)$ on a sample $z = (x, y)$. The samples are supplied by a set $\mathcal{A}$ of *Agents*, with agent i providing point $z_i = (x_i, y_i)$. We will denote by $\mathcal{A}_{-i}$ the set of agents excluding agent i. Given a set of data $Z = \{z_i\}_{i=1}^n$, the empirical risk is $R(Z, \theta) = \frac{1}{n}\sum_i L(z_i, \theta)$. The center uses a scoring function $s(z)$ to determine the reward it pays for the data z.

Our approach differs from other papers on this topic in that we make relatively few assumptions about agent beliefs and effort models. We consider the agent and center models as follows:

Agent: It is assumed that each agent i must exert effort $e_i(o)$ to observe data point o. We adopt a simple effort model, in which the agent makes a binary choice either to exert effort to make an observation, or exerts 0 effort and reports a data point based on some *heuristic*. The observation o_i and the necessary effort expended are unknown a priori to the agent. When the agent decides to make an observation, there is some expected effort δ_i over the distribution of observable data points, which is known to the agent a priori. This value can vary amongst the agents. An observation $o_i = g(\phi_i|\psi_i)$, where ϕ_i is drawn from some shared underlying distribution Φ, ψ_i is a hidden latent random variable drawn from a shared distribution Ψ, and g is a function that applies noise to ϕ_i given ψ_i. The agents have 2 beliefs regarding their noise, which we formulate as follows:

- Unbiased: $\forall \phi \in \Phi, \forall i, \mathbb{E}_\Psi[g(\phi|\psi_i)] = \phi$
- Non-Trivial: $p_\Psi(g(\phi|\psi_i) = \phi) = 0$

The Center will employ a scoring function $s(\cdot)$ to provide payments to the Agents, dependent on their reports; we postpone any details about this scoring function until the next section. We assume that agents possess no prior belief about Φ or about the model used by the center, and therefore there is no belief update process when making an observation o_i. This means that the strategy space of the agents is limited to the following:

- Expend effort δ_i to make observation o_i and make the truthful report $r_i = o_i$, receiving expected utility $s(o_i) - \delta_i$, where s is the scoring function used by the center. We refer to this as the *truthful strategy*.
- Choose some arbitrary heuristic H_i and report $r_i = h_i$, with $h_i \in H_i$, expending 0 effort and receiving utility $s(h_i)$.
- Opt-out and not observe or report anything.

Agents are rational, so they will choose the strategy that maximizes their expected utility. We make one further assumption about agent beliefs. For this we introduce the notion of *risk-monotonicity*, which is the notion that a model learner is monotonic in the true risk over the number of data points in the training set. While (Loog, Viering, and Mey, 2019) show that not all empirical risk minimizers are risk-monotonic in the number of training points, their counter-examples are adversarially constructed. As Agents have no prior information about the distributions, we consider it reasonable to make the following formal assumption:

Agents believe the center's model is risk-monotonic with respect to the true distribution Φ, i.e. an agent expects that a point drawn from Φ will not worsen the model's expected risk when evaluated on Φ.

Center: The center wishes to construct a model because they believe they can extract some profit from this model. We assume the profit is a function $f(R)$ of the model risk. The expected utility of the center is then the profit $f(R) - c(R)$, where $c(R)$ is the expected cost of constructing a model with risk R. We do not assume a priori that c is known to the center, although it may be computable depending on the choice of scoring function s. We assume that $f(R)$ is monotonically decreasing. In order to evaluate the model risk R, we assume two cases: *(a)* the Center may possess an *independent test set*, or *(b)* it may have to *acquire a test set* by randomly selecting reports from the agents with some probability p for each agent.

Implications: We contrast our setting with that considered by Cai, Daskalakis, and Papadimitriou [2015]. In this paper presents a mechanism which forms a Dominant Strategy Equilibrium (DSE), but acquiring such a strong solution concept requires the adoption of certain strong assumptions. The authors assume that each agent chooses an *effort level*, and the variance of the accuracy of their reports is a strictly decreasing convex function of that effort. Furthermore, these functions need to be exactly known to the Center. Finally, the paper does not consider the strategy space of non-truthful heuristic reporting. In this paper, we only require that the cost of effort is bounded. Furthermore, our strategy space is more expressive in the sense that, as in real-life scenarios, data providers can choose which data to provide and not just which effort level to exert.

It is easily provable that the strong solution concept of a truthful DSE is impossible to achieve in general with our more flexible assumptions. Suppose there is a mechanism that has a truthful DSE for some binary observation signal. Then suppose for the same mechanism the observation signal is inverted, i.e. the agents observe 0s as 1s and vice versa. From the perspective of any agent, if all other agents report the inverse of the observation, i.e. the original un-inverted signal, this agent is clearly incentivized to also report the inverse of the observation by the assumption that truthful report of the original signal is a DSE. Hence, there does not exist a truthful DSE for this inverted signal.

3. Game-theoretic Incentives

An *incentive scheme* is a function that maps data points z_i to payments $s(z_i)$; intuitively, a good incentive scheme should overcome the cost of effort (as otherwise agents are not incentivized to submit any observations) but also, crucially, to reward based on the effect that the data point z_i has on improving the accuracy of the trained model. For this reason, we will design incentive schemes via the use of influences. Let $Z_{-j} = \{z_i\}_{i \neq j}$ and let

$$\hat{\theta} = \arg\min_{\theta} R(Z, \theta) \quad \text{and} \quad \hat{\theta}_{-j} = \arg\min_{\theta} R(Z_{-j}, \theta).$$

We will assume that the Center is in possession of an *test set* $T = \{z_k\}$. Then the *influence* of z_j on the test set is defined as

$$\text{infl}(z_j, T, \theta) = R(T, \hat{\theta}_{-j}) - R(T, \hat{\theta}).$$

We will simply write $\text{Infl}(z_j)$, when T and θ are clear from the context. Then, we can design the following incentive scheme:

- Case 1: The center possesses an independent test set: $s(r_i) = \alpha_c \text{Infl}(r_i) - \epsilon$, where $\epsilon > 0$ is a very small value.
- Case 2: The center draws its test set from the reports, with each report being added to the test set with some probability p: $s(r_i) = \frac{\alpha_c \text{Infl}(r_i) - \epsilon}{(1-p)}$ or 0 if the report is added to the test set.

We will discuss how the center can choose α_c in detail.

Following standard game-theoretic terminology, we will say that an agent supplying point r_j is *best responding* to the set of strategies r_{-j} chosen by the other agents, if the strategy that it has chosen maximizes the quantity $\mathbb{E}[s(r_j|r_{-j}) - e_i(r_j)]$ over all possible alternative reports r'_j, where the expectation is over the distribution of reports of the other agents. We will say that a vector of strategies (i.e., a strategy profile) $(r_1, \ldots, r_n)$ is a *Bayes-Nash equilibrium (BNE)* if, for each agent j, r_j is a best response. If r_j is a best response to any set of strategies of the other players, we will say that r_j is a *dominant strategy*.

3.1. Incentives for Agents

For the lemmas in this section, we make the following assumptions:

- Observation noise is unbiased and non-trivial, as stated in the previous section.
- Agents have no prior knowledge of the true distribution Φ or the model of the center.

The proofs of the following statements are omitted due to lack of space, but are included in the supplement.

Lemma 1. *An agent having made no observation believes the expected influence of any particular report to be 0.*

Lemma 2. *An agent A_i believes that, almost certainly, given a finite number of reports, $\mathbb{E}_{\Phi}[\mathbb{E}_{\Psi}[\textit{Infl}(o_i|\{o_j\}_{j \neq i})]] > 0$ when evaluated on $\{z_{test}\}$ with z_{test} in the distribution of observations.*

The following theorem asserts that as long as the test set consists of truthful information, the agents have a dominant strategy of either being truthful of opting out. In the case where the Center possesses an independent test set, this condition is satisfied trivially. In addition, the theorem provides a BNE guarantee: even if the test set is assembled by the Agents themselves, they have incentives to be truthful (or opt out) as long as the other Agents are truthful. However, it might still be the case that certain Agents report heuristically; we consider this case in Subsection 4.4 and show robustness of our incentives schemes there.

Theorem 3. *Suppose that (a) the noise is unbiased and non-trivial, (b) the agents do not have knowledge of the distribution or the model and (c) the test set consists of truthful information. Then, there is a large enough α_c such that every agent, almost certainly, has a dominant strategy of either being truthful or opting-out.*

3.2. Incentives for the Center

We now consider how this incentive scheme relates to the utility of the center. Given a center with profit function $f(R)$, under a truthful BNE, the utility of the center is at least:

$$f(R) - \sum_i \alpha_c \text{Infl}(o_i)$$

But we require that $\alpha_c > \frac{\max(\delta_i)}{\mathbb{E}_\Phi[\mathbb{E}_\Psi[\text{Infl}(o_i|\{o_j\}_{j\neq i})]]}$. Therefore, we have that the utility of the center in expectation is $f(R) - n\max(\delta_i) - \epsilon$ where n is the number of agents. In order for the center to have positive utility, the following must be satisfied: $\max(\delta_i) < \frac{f(R)}{n}$. It is possible in a real-world case that this inequality is not satisfied. This represents the case when the labor of the agents is too expensive to scale appropriately with the profit that the center would receive from building the model. However, a center can manipulate the value of α_c in order to attain a positive utility as long as not too many agents are forced to opt out.

4. Practical Setting

In this section, we present methods that allow this influence-based incentive scheme to be used in a practical setting. We consider in particular the following cases:

- Learning the distribution of a random variable with k values: an Agent i reports its own instance of the variable $z_i = x_i$, and the model $\hat{\theta}$ is an estimate of a probability distribution $\theta_i = p(x_i)$. Each model parameter is the normalized frequency of reports of value x_i.
- Learning a linear regression model: an Agent i reports a pair $z_i = (\underline{x_i}, y_i)$, and the Center learns a regression model $\hat{y} = \underline{\theta} \cdot \underline{x} + \theta_0$ with a least squares loss function.
- Learning a logistic regression model: as above, but with logistic loss function.

4.1. Learning Distributions

This setting describes the classical crowdsourcing setting for the Peer Truth Serum (Faltings, Jurca, and Radanovic, 2017). The Agents provide the Center with their observations of different instantiations of a discrete random variable, and the Center constructs a model of the variable's distribution as a histogram of the observations reported by the Agents. Specifically, we assume that the Center elicits an unknown distribution of m values $p_1, .., p_m$. Agents observe and report samples taken from the distribution and the center aggregates them in a histogram. We measure time by the number n of values collected by the center.

Initially, each histogram cell $h_j, j \in [1, .., n]$ is set to 1, corresponding to a uniform estimated probability $\hat{\theta}_j = 1/m$. Upon receiving a report of value x_i, the Center updates the model from $\hat{\theta}_{-i}$to $\hat{\theta}$ using the following update rule:

$$\hat{\theta}(x_i) = \hat{\theta}_{-i}(x_i) + (1 - \hat{\theta}_{-i}(x_i)) \cdot \delta = \delta + \hat{\theta}_{-i}(x_i) \cdot (1 - \delta)$$
$$\hat{\theta}(x_j) = \hat{\theta}_{-i}(x_j) \cdot (1 - \delta) \text{ for } x_j \neq x_i$$

The Center uses the *logarithmic scoring rule* (LSR) as a loss function to measure the quality of the model for predicting a test data point z:

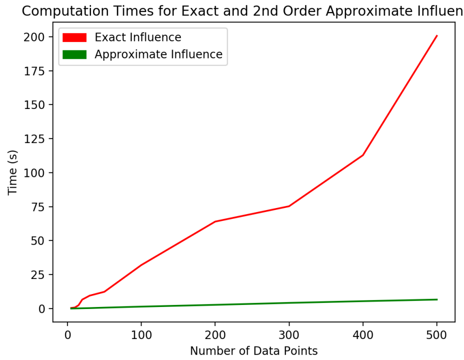

Figure 2: The exact influence is shown to become computationally prohibitive for logistic regression with only a moderate number of data points, while the computation time for the approximate influence increases relatively slowly.

$$LSR(\underline{\hat{\theta}}, z) = \ln \hat{\theta}(z)$$

We could compute the influence of a new data point on the loss function directly. However, it is instructive to consider an approximation using the Taylor expansion of the loss function. *Due to lack of space, me refer to the supplement for the details.*

From there, we can derive the following payment function:

$$\text{infl}(x_i, z) = \delta(n) \left(\frac{1}{l} \sum_z \frac{\mathbf{1}_{x_i = z}}{\hat{\theta}(x_i)}_{-i} - 1 \right)$$

where l is the number of test data points, δ is a scaling factor, for example $\delta(n) = 1/(n+1)$. If the values don't match, the agent has to pay $\delta(n)$ (which could be a participation fee charged up front). We note that when $\delta(n)$ is a constant, for example $\delta(n) = 1$, this scheme exactly matches the Peer Truth Serum (Faltings, Jurca, and Radanovic, 2017) and has been studied extensively in the literature.

4.2. Regression Learning

4.2.1. Influence Approximation

Trying to practically implement an incentive-based payment mechanism for regression learners imposes a host of challenges. The first is the computational cost of computing the influence for an agent. Specifically, we must compute $\hat{\theta}_{-i}$, which would involve entirely retraining the model. We present an approximation method based on the method described in (Koh and Liang, 2017), which uses the first term of the Taylor expansion, like in Section 4.1 (see supplement).

$$\text{infl}(z_{test}, z_j) = \frac{1}{n} \nabla_\theta L(z_{test}, \hat{\theta}) H_\theta^{-1} \nabla_\theta L(z_j, \hat{\theta})$$

This approximation, however, has the undesirable property that the mean influence of the training data is 0, as shown by the definition of $\hat{\theta}$ as the solution to $\sum \nabla_\theta L(z_j, \hat{\theta}) = 0$. As the sum of influences of the data points is in general positive, this approximation is insufficiently precise. We therefore include the 2nd order term in the Taylor expansion of the empirical risk. Let $\partial\theta_j$ be the change in theta due to up-weighting a training point z_j, and let H_i be the Hessian computed only on z_i.

$$\partial\theta_j = \frac{1}{n} H_\theta^{-1} \nabla_\theta L(z_j, \hat{\theta}) + \frac{1}{n^2} H_\theta^{-1} H_j H_\theta^{-1} \nabla_\theta L(z_j, \hat{\theta})$$

Also taking into account the second order approximation of the change in the loss on a test point when computing the influence:

$$\text{infl}(z_{\text{test}}, z_j) = \left(\nabla_\theta L(z_{\text{test}}, \hat{\theta}) + \frac{1}{2} H_{\theta, z_{\text{test}}} \cdot \partial\theta_j \right) \cdot \partial\theta_j$$

Experimental Results We ran simulations to confirm the improved accuracy of the 2nd order approximation method and to demonstrate its computational efficiency. For the case of linear regression, computing the exact influence for each data point can be computationally feasible, but with a high enough input dimension the approximation will be more computationally efficient. For a model that is learned via an SGD method, such as a logistic regressor, it is clear that the influence approximation will provide significant improvements, as shown in Fig. 2. For all of our other simulations, we used an artifically generated linear dataset with noise, along with 4 datasets from the UCI database with a linear regression model: ”Red and White Wine” (Cortez et al., 2009), ”Air Quality” (De Vito et al., 2008), ”Crime” (Redmond and Baveja, 2002), and ”Parkinsons” (Tsanas et al., 2009). Non-predictive fields, redundant fields, and fields with many missing values were removed.

Using 1000 points for training and 200 for testing, we evaluated the exact influence, 1st order approximation and 2nd order approximation for each data point, recording the L1 and L2 norms between the approximations and the exact influence. We then evaluated the worst case and average improvement factors for the 2nd order approximation over the 1st order approximation (2nd order error / 1st order error). The worst case improvement for the L1 norm was 0.410, with the average being 0.0789 (lower is better). The worst case improvement for the L2 norm was 0.482, with the average being 0.0821. This means that for both the L1 and the L2 norms, the 2nd order approximation was on average about 12 times as accurate as the 1st order approximation. We also report the means of the L1 and L2 norms between the 2nd order approximation and the exact influence to demonstrate that it is indeed accurate: $1.16 * 10^{-3}$ for L1 and $2.45 * 10^{-5}$ for L2.

4.2.2. Batch Processing

In a practical implementation, data arrives sequentially. Prior, we assumed that the influences would be computed over the entire dataset once all the data has been collected. Ideally, the Center could compute the influence and provide the payment immediately when a data point arrives. This has the advantage of allowing the Center to perform a priori budgeting. Suppose we have a dataset $\{z_i\}$ such that i indicates the time of arrival of each datapoint. Then the sum of influences is the overall change in risk on the test set T.

$$\sum_{j=0}^{n} R(T, \hat{\theta}_{\{z_i\}_{i<j}}) - R(T, \hat{\theta}_{\{z_i\}_{i\leq j}}) = R(T, \hat{\theta}_\emptyset) - R(T, \hat{\theta}_{\{z_i\}})$$

By assigning a utility to the overall change in risk, the Center can budget the entire data collection period before implementing the mechanism. However, computing the influence for each data point as it arrives can be computationally prohibitive, even using the influence approximation. The computation time of the approximation, in terms of complexity, is dominated by computing H^{-1}, which must be computed every time the model is updated. The Center can strike a balance between the two extremes by grouping the data into batches, such that H^{-1} is only computed once per batch.

With respect to a single batch, the game theory is the same as the one-batch case, however, we must now consider how batch processing affects incentives with respect to the time of reporting. We observe the following: the 1st order influence approximation presented in (Koh and Liang, 2017) has absolute error with respect to the exact influence of $O(\frac{1}{n^2})$, and is 0 in expectation. Therefore, in expectation, the exact influence is $O(\frac{1}{n^2})$. With this, it is clear that batch processing incentivizes Agents to report as early as possible, which is a desirable property for the Center.

4.2.3. Initializing the Model

One point we did not address was the meaning of $\hat{\theta}_\emptyset$ in Section 4.2.2: the optimal parameters of the model given no training dataset. In many cases, linear regression for example, this is not properly defined. We term this the *initialization of the model*. We interpret this *initial model* as the aggregate knowledge of the Center prior to the data collection period. If the Center already has some data it can use for training, then it won't need to collect as many data points or spend as much money during the data collection period. If the Center, rather than having a prior dataset, has some knowledge about the distribution of the model it wishes to learn, it can artificially generate a dataset by sampling from this prior distribution, with the number of samples corresponding to the confidence of the Center in the prior model. In the case where the Center has no prior knowledge, we consider this from an information theoretic perspective to be a state of maximum entropy, and therefore the initial model would be determined by sampling from the uniform distribution within the appropriate bounds.

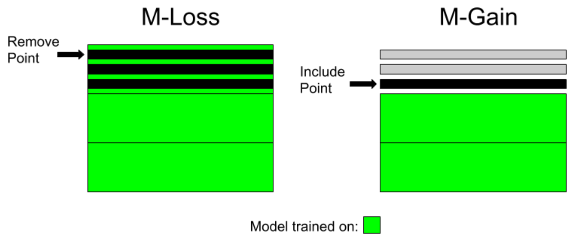

Figure 3: M-Loss is trained on all points in current batch, with influence computed by removing a point. M-Gain is trained on all prior batches, with influence computed by adding a point from current batch.

4.3. M-Loss and M-Gain

With batch processing, the Center has two choices in how to implement the mechanism. The Center may include the most current batch in updating the model and compute the influence of each data point as though it were removed, or it could exclude the current batch and compute the influence of each data point as though it were added to the rest. We call these two methods *M-Loss* and *M-Gain* respectively, as shown in Fig. 3. It is clear by construction that for a batch size of one, these two methods are equivalent, and the sum of influences is equal to the overall change in risk. For the sake of computational efficiency, the Center will want to choose a batch size greater than 1. We note that M-Loss will underestimate the expected influence in the 1-batch case because the influence of points that arrive early in the batch won't be computed until the later points arrive. Similarly, we observe that M-Gain will overestimate. We wish to characterize the extent to which M-Loss and M-Gain underestimate and overestimate respectively, so the Center can compensate. We restrict ourselves to the case of linear regression, but the analysis can be extended to any model in which the optimal parameters have a closed-form solution.

Let us consider two probability distributions Φ_1 and Φ_2, and we assume they describe an input-output relationship such that $\Phi(x, y) = q(x)p(y|x)$, and $q_1(x) = q_2(x)$. This assumption merely asserts that the data we are collecting is drawn from the same domain regardless of the distribution of the output. Distributions Φ_1 and Φ_2 determine, in expectation, models M_1 and M_2 respectively. Let us now define $R_{i,j}$ as the expected risk of model M_i evaluated on distribution Φ_j. Using the standard mean-squared-error loss function, we have that $R_{i,j} = R_{j,j} + \mathbb{E}[(M_i - M_j)^2]$. Now suppose we sample N_1 points from Φ_1 and N_2 points from Φ_2 to form our training set $\{z\}$. Because the linear regression solution is linear with respect to y, and $q(x)$ is fixed, then $\{z\}$ determines in expectation a model $M_c = \frac{N_1 M_1 + N_2 M_2}{N_1 + N_2}$. With this, let us consider the practical application where Φ_1 is the initialization distribution and Φ_2 is the distribution of reports from the Agents. Then when we evaluate the model, we are only concerned with the error of the mixed model M_c evaluated on Φ_2:

$$R_{c,2} = R_{2,2} + \left(\frac{N_1}{N_1 + N_2}\right)^2 \mathbb{E}\left[(M_2 - M_1)^2\right]$$

To simplify, we fix $N_1 = Q$ as the number of points used for initialization, we define $r = \mathbb{E}[(M_2 - M_1)^2]$, and we let N_2 vary as x. Then we have our expected empirical risk in terms of x:

$$R(x) = \frac{Q^2 r}{(Q + x)^2} + R_{2,2}$$

We can approximate the influence of a data point arriving after x data points as the negative of the derivative of the risk:

$$-\frac{\partial R}{\partial x} = \frac{2Q^2 r}{(Q + x)^3}$$

Now we consider batch size b. We can compute the expected overall change in loss of some arbitrary batch k, with k indexing starting at 1.

$$\Delta R_b(k) = R((k-1)b) - R(kb) = \frac{bQ^2 r(2Q + (k-1)b)^2}{(Q + (k-1)b)^2 (Q + kb)^2}$$

Now we consider the sum of influences of points in batch k for M-Loss and M-Gain.

$$S_{\text{mloss},b}(k) = -b\frac{\partial R}{\partial x}\bigg|_{kb} = \frac{2bQ^2 r}{(Q + kb)^3}$$
$$S_{\text{mgain},b}(k) = -b\frac{\partial R}{\partial x}\bigg|_{(k-1)b} = \frac{2bQ^2 r}{(Q + (k-1)b)^3},$$

Comparing these to the change in risk, we get the following ratios:

$$D_{\text{mloss},b}(k) = \frac{S_{\text{mloss},b}(k)}{\Delta R_b(k)} = \frac{2(Q + (k-1)b)^2}{(Q + kb)(2Q + (2k-1)b)}$$
$$D_{\text{mgain},b}(k) = \frac{S_{\text{mgain},b}(k)}{\Delta R_b(k)} = \frac{2(Q + kb)^2}{(Q + (k-1)b)(2Q + (2k-1)b)}$$

By computing these values, the Center can pick an arbitrary batch size and divide the influence scores by these formulas such that the expected sum of influences is equal to the overall change in risk, as in the case of batch size 1. We note that these formula have constant growth rate with respect to the number of points $Q + kb$ and they asymptotically approach the constant function $D_b(k) = 1$. Therefore, dividing the influence scores by these formulas will not affect the incentive for early reporting.

We note that this analytic method only applies to linear regression, and that it can only be reasonably extended to learners with closed-form solutions for the optimal parameters. However, the center can approximate this method by using the observed influences and change in risk across a batch and rescaling with the ratio of these two empirical values, rather than the a-priori expected ratio.

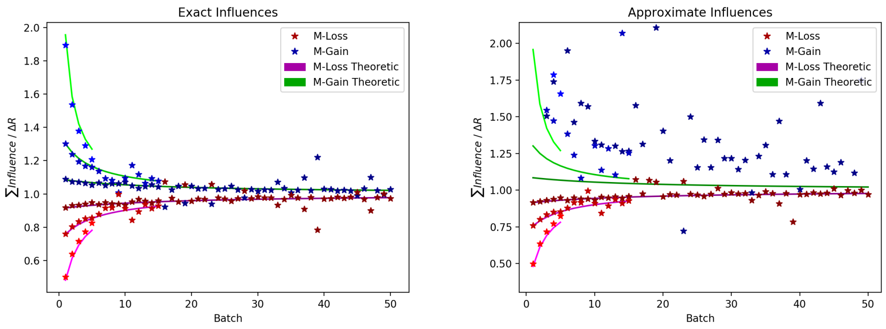

Figure 4: Ratio between Sum of Influences and Change in Loss with respect to batch size.

4.3.1. Experimental Results

We present experimental results to demonstrate the validity of the rescaling formula in real scenarios. We ran simulations, using the same datasets as in Section 4.2.1, to estimate the effect of the batch size on the ratios D_{mloss} and D_{mgain}. We ran each simulation with 1500 total training points with a varying batch size. Given a fixed batch size, we ran 10 trials for every dataset and aggregated them to form a more general estimate of S_{mloss}, S_{mgain}, and ΔR. We then took the ratios of these aggregates and compared against our theoretical results for D_{mloss} and D_{mgain} in Fig. 4. We ran this same simulation with different numbers of initial points 20, 100, 200, and 500. We have chosen only to show the case with 500 initial points, although the other simulations show the same relationship. Each line represents a different batch size. We have chosen to plot batch sizes 30, 100, and 300 for ease of visualization.

4.4. Using agent reports as test data

We have shown in Theorem 3 that under some reasonable assumptions, truthful reporting forms a BNE for the agents. However, this requires a truthful test set, which might not always be at the disposal of the Center. We will show that, even if we collect the reports as test data and a subset of agents report using the same heuristic strategy (for example, reporting according to some gaussian distribution), under certain conditions the heuristic reporters will still be unprofitable. We can use the same analysis as in Section 4.3. Once again we consider two distributions Φ_1 and Φ_2 with the same assumptions as before. Let Φ_1 be the distribution of heuristic reports and Φ_2 by the distribution of truthful reports. Suppose that we draw x_1 points from Φ_1 and x_2 points from Φ_2, with $n = x_1 + x_2$. Then we have our mixed model $M_c = (x_1 M_1 + x_2 M_2)/n$. The mixed model will then be evaluated on the mixed distribution $\Phi_c = (x_1\Phi_1 + x_2\Phi_2)/n$. Taking the empirical risk $R_{c,c}$, we compute the influences:

$$-\frac{\partial R_{c,c}}{\partial x_1} = \frac{p}{n}R_{1,1} - \frac{p}{n}R_{2,2} - \frac{p(2p-1)r}{n}$$

$$-\frac{\partial R_{c,c}}{\partial x_2} = -\frac{1-p}{n}R_{1,1} + \frac{1-p}{n}R_{2,2} + \frac{(1-p)(2p-1)r}{n}$$

We wish to know under what conditions the agents reporting according to Φ_2 will receive a higher reward than those reporting according to Φ_1:

$$0 < -\frac{\partial R_{c,c}}{\partial x_2} - -\frac{\partial R_{c,c}}{\partial x_1}$$

This yields for the fraction $p = x_2/n$ of non-heuristic reports:

$$p > \frac{1}{2} + \frac{R_{2,2} - R_{1,1}}{2r}$$

Because $R_{i,i}$ represents the risk of model i evaluated on the points used to compute model i, we will call it the residual risk of the model. This bound shows that the problematic case for the Center is when the residual risk is lower for the heuristic model than the truthful model. In this case, the Center will require more than a majority of truthful reporters to maintain proper truthful incentives. If the heuristic reporters are uncoordinated or random, causing the residual risk in the truthful model to be lower than that of the heuristic model, then the Center does not even require a majority of truthful reporters to make heuristic reporting unprofitable. If the heuristic model is significantly different from the truthful model, the the bound will tend towards $\frac{1}{2}$, so profitability will be decided by the majority.

5. Conclusion

We have presented a novel incentive scheme for good quality data acquisition based on influence functions. The influence score has the clear advantage that the Center will only have to pay for data that improves the performance of the model. We have shown for regression models how influence can be approximated and processed in batches for efficiency, and developed a theory that allows correcting the difference between the influence and the overall change in loss.We also analyzed the influence of agents that report heuristically without making the effort to collect actual data and showed that the mechanism is robust to irrational reports.

References

Cai, Y.; Daskalakis, C.; and Papadimitriou, C. 2015. Optimum statistical estimation with strategic data sources. In Grünwald, P.; Hazan, E.; and Kale, S., eds., *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, 280–296. Paris, France: PMLR.

Caragiannis, I.; Procaccia, A.; and Shah, N. 2016. Truthful univariate estimators. In *International Conference on Machine Learning*, 127–135.

Chen, Y.; Immorlica, N.; Lucier, B.; Syrgkanis, V.; and Ziani, J. 2018a. Optimal data acquisition for statistical estimation. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, 27–44. ACM.

Chen, Y.; Podimata, C.; Procaccia, A. D.; and Shah, N. 2018b. Strategyproof linear regression in high dimensions. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, 9–26. ACM.

Cook, R. D., and Weisberg, S. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics* 22(4):495–508.

Cortez, P.; Cerdeira, A.; Almeida, F.; Matos, T.; and Reis, J. 2009. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems* 47(4):547–553.

Cummings, R.; Ioannidis, S.; and Ligett, K. 2015. Truthful linear regression. In *Conference on Learning Theory*, 448–483.

De Vito, S.; Massera, E.; Piga, M.; Martinotto, L.; and Di Francia, G. 2008. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. *Sensors and Actuators B: Chemical* 129(2):750–757.

Dekel, O.; Fischer, F.; and Procaccia, A. D. 2010. Incentive compatible regression learning. *Journal of Computer and System Sciences* 76(8):759–777.

Faltings, B., and Radanovic, G. 2017. Game theory for data science: eliciting truthful information. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 11(2):1–151.

Faltings, B.; Jurca, R.; and Radanovic, G. 2017. Peer truth serum: Incentives for crowdsourcing measurements and opinions. *CoRR* abs/1704.05269.

Jia, R.; Dao, D.; Wang, B.; Hubis, F. A.; Hynes, N.; Gurel, N. M.; Li, B.; Zhang, C.; Song, D.; and Spanos, C. 2019. Towards efficient data valuation based on the shapley value. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Koh, P. W., and Liang, P. 2017. Understanding black-box predictions via influence functions. In Precup, D., and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 1885–1894. International Convention Centre, Sydney, Australia: PMLR.

Konečnỳ, J.; McMahan, H. B.; Yu, F. X.; Richtárik, P.; Suresh, A. T.; and Bacon, D. 2016. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

Loog, M.; Viering, T.; and Mey, A. 2019. Minimizers of the empirical risk and risk monotonicity. In *Advances in Neural Information Processing Systems*, 7476–7485.

Meir, R.; Procaccia, A. D.; and Rosenschein, J. S. 2012. Algorithms for strategyproof classification. *Artificial Intelligence* 186:123–156.

Perote, J., and Perote-Pena, J. 2004. Strategy-proof estimators for simple regression. *Mathematical Social Sciences* 47(2):153–176.

Radanovic, G.; Faltings, B.; and Jurca, R. 2016. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology (TIST)* 7(4):48.

Redmond, M., and Baveja, A. 2002. A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research* 141(3):660–678.

Shah, N.; Zhou, D.; and Peres, Y. 2015. Approval voting and incentives in crowdsourcing. In *International Conference on Machine Learning*, 10–19.

Tsanas, A.; Little, M. A.; McSharry, P. E.; and Ramig, L. O. 2009. Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests. *IEEE transactions on Biomedical Engineering* 57(4):884–893.

Vuurens, J.; de Vries, A. P.; and Eickhoff, C. 2011. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *Proc. ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR'11)*, 21–26.

Westenbroek, T.; Dong, R.; Ratliff, L. J.; and Sastry, S. S. 2017. Statistical estimation with strategic data sources in competitive settings. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 4994–4999. IEEE.

Westenbroek, T.; Dong, R.; Ratliff, L. J.; and Sastry, S. S. 2019. Competitive statistical estimation with strategic data sources. *arXiv preprint arXiv:1904.12768*.