Collaborative and Privacy-Preserving Machine Teaching via Consensus Optimization

Yufei Han[1], Yuzhe Ma[2], Christopher Gates[1], Kevin Roundy[1], and Yun Shen[1]

[1]Symantec Research Labs
[2]University of Wisconsin-Madison

***Abstract*—In this work, we define a collaborative and privacy-preserving machine teaching paradigm with multiple distributed teachers. We focus on consensus super teaching. It aims at organizing distributed teachers to jointly select a compact while informative training subset from data hosted by the teachers to make a learner learn better. The challenges arise from three perspectives. First, the state-of-the-art pool-based super teaching method applies mixed-integer non-linear programming (MINLP) which does not scale well to very large data sets. Second, it is desirable to restrict data access of the teachers to only their own data during the collaboration stage to mitigate privacy leaks. Finally, the teaching collaboration should be communication-efficient since large communication overheads can cause synchronization delays between teachers.**

To address these challenges, we formulate the collaborative teaching as a consensus and privacy-preserving optimization process to minimize teaching risk. We theoretically demonstrate the necessity of collaboration between teachers for improving the learner's learning. Furthermore, we show that the proposed method enjoys a similar property as the Oracle property of adaptive Lasso. Empirical study illustrates that our teaching method can deliver significantly more accurate teaching results with high speed, while the non-collaborative MINLP-based super teaching becomes prohibitively expensive to compute.

I. Introduction

Machine teaching [1]–[5] studies the inverse problem of machine learning, where a teacher already has a specific target model (θ^*) it wants to teach some other student (learner), and the teacher designs the optimal training set such that the student can efficiently learn the target model. The constructed training set does not need to be independent and identically distributed. The teacher is allowed to design any instance in the input space, which enables flexibility when generating an efficient training set. The efficacy of teaching is measured by computational cost of the model training, accuracy of the derived learner, robustness of the training process and so on. In general, machine teaching places a strong emphasis on the teacher and its power to control data. Machine teaching is connected to machine learning fundamentals as it defines abstractions and interfaces between the learning algorithm and the teaching paradigm. Research on machine teaching has not only great theoretical value, but also applications in personalized education and human-in-the-loop learning.

Super teaching is an interesting phenomenon unveiled recently in machine teaching. As stated in [6], a learner is super-teachable if a teacher can trim down an *i.i.d.* training set while enhancing the learning performance. Distinct from training set reduction where the target model is hidden from the learner, super teaching assumes the teacher knows the target model and rely on such knowledge to select a training subset so that a student learner can learn better on that subset.

Prior work in super teaching assumes that only one central teacher is present and it has full knowledge of all data instances used for teaching. As the privacy concern of data security becomes increasingly important, research in super teaching faces the following important challenges. First, instead of transferring all data used to teach to the central teacher where teaching is conducted, it is preferable to keep data on the local devices and conduct teaching with multiple distributed teachers. Each teacher only accesses the data samples hosted by itself. It avoids heavy overheads of the data transferring and prevents leaks of private information contained in the local data set. However, it is not clear whether organizing a consensus collaboration between teachers can provide merits of teaching compared to independently conducting teaching by each teacher in a stand-alone mode. Little efforts have been devoted to discuss how to organize an efficient collaborative super teaching paradigm to achieve good teaching performances and a privacy-preserving process of teaching at the same time. Furthermore, the state-of-the-art super teaching method proposed in [6] is formulated as a mixed-integer non-linear programming (MINLP) problem. The general computational complexity of MINLP problem is undecidable in theory [7]. In the worst case, the popular heuristic solver, such as Branch-and-Bound (BnB) method, has an exponential time complexity thus becomes prohibitively expensive given large-scale training data. Solving a MINLP problem with distributed players is even more difficult, as it usually needs a central processor to allocate the resource across multiple players to solve the overall problem [8]. Therefore, the central processor can access the local private data, which potentially violates the privacy regulation. Besides, frequent synchronization between the central process and the end-devices can cause severe latency given a low-communication environment. Finally, limited computing capability of end-devices in a distributed network can not afford to the intensive computation of solving the MINLP problem.

We propose a novel computationally efficient distributed super teaching paradigm, which coherently facilitates collaboration between multiple teachers in a privacy-preserving way. Our study confirms a well-known intuition: ***A carefully or-***

ganized consensus collaboration between different teachers can enhance the teaching performances. We also show that independently conducting teaching in a non-colluded way can even make the teaching performance deteriorate. Furthermore, the privacy-preserving design of the proposed collaborative teaching paradigm encourages information sharing between teachers in the collaboration stage.

II. Related work

A. Machine Teaching

Machine teaching was originally proposed in [1], [2]. It has attracted plenty of research interest, most of which focus on studying a quantity called the teaching dimension, i.e., the size of the minimal training set that is guaranteed to teach a target model to the student. For example, [1] provides a discussion on the teaching dimension of version space learners, [9] analyzes the teaching dimension of linear classifiers, and [4] studies the optimal teaching problem of Bayesian learners. In standard machine teaching, the student is assumed to passively receive the optimal training set from teacher. Later works consider other variants of teaching setting, e.g., in [3], [10], the student and the teacher are allowed to cooperate in order to achieve better teaching performance. More studies of machine teaching can be found in [5], [11]–[14].

Machine teaching as a theoretical regime also has many applications in cognitive science and computer security. One application is the personalized education, where a clairvoyant teacher can help design minimal teaching curriculum for a human student such that an educational goal is achieved [15]. As another popular application, machine teaching can be used to perform data poisoning attacks of real-world machine learning systems [16]–[18]. In such cases, the teacher is viewed as a nefarious attacker who has a specific attack goal in his mind, while the student is some machine learner, then the teaching procedure corresponds to minimally tampering the clean dataset such that the machine learner learns some sub-optimal target model on the corrupted training set.

Instead of artificially designing the training set, super teaching [6] selects a subset from an *i.i.d.* training set to conduct teaching. Mathematically, super teaching is defined as below.

Definition 1 (Super Teaching)**.** *Let S be an n-item iid training set, and T be a teacher who selects a subset $T(S) \subset S$ as the training subset for learner A. Let $\hat{\theta}_S$ and $\hat{\theta}_{T(S)}$ be the model learned from S and $T(S)$ respectively. Then T is a super teacher for learner A if $\forall \delta > 0, \exists N$ such that $\forall n \geq N$*

$$\mathbf{P}_S \left[R(\hat{\theta}_{T(S)}) \leq c_n R(\hat{\theta}_S) \right] > 1 - \delta, \tag{1}$$

where R is some teaching risk function, the probability is with respect to the randomness of S, and $c_n \leq 1$ is a sequence called the super teaching ratio.

The idea of selecting an informative training subset is also explored in [19]. In the proposed *learning-to-teach* framework, the teacher conducting subset selection is modeled with Deep Neural Nets (DNN). The goal of teaching is to select training samples to make faster convergence of the DNN based learner. The teacher network is tuned via reinforcement learning with reward signals encouraging fast descent of the classification loss of the learner. In contrast to learning-to-teach, super teaching in [6] focuses on a more general teaching goal, which drives the student to learn the expected model. Although only simple learners such as Logistic Regression are considered in [6], the theoretical study over the teachability of super teaching can be further applied to many advanced learners.

Inspired by the teachability theory proposed in [6], our work extends the horizon of super teaching by studying applicability of a collaborative and privacy-preserving teaching scenario. Different from learning-to-teach, multiple teachers are present as collaborative players in the teaching activity. Furthermore, training data hosted by any one teacher can not be accessed by the others in our problem setting, whereas learning-to-teach assumes the teacher network can access all the training data.

B. Federated Learning

Another relevant branch of research is federated learning [20]. Federated learning is a communication-efficient and privacy-preserving distributed model training method over distributed agents. Each agent hosts their own data instances and is capable of computing local model update. In each round of model training, the training process is first conducted on each node in parallel without inter-node communication. Only the local model updates are aggregated on a centralized parameter server to derive the global model update. The aggregation is agnostic to data distribution of different agents. Neither the centralized server, nor the local agents have visibility of the data owned by any specific agent. In [21], a communication-efficient distributed optimization method named *CoCoA* is proposed for training models in a privacy-preserving way. *CoCoA* applies block-coordinate descent over the dual form of the joint convex learning objective and guarantees sub-linear convergence of the federated optimization. Furthermore, the optimization process does not require to access data instances hosted by each node. Only local dual variable updates need to transfer from local nodes to the central server. This property makes *CoCoA* inherently appropriate for federated training.

A federated data poisoning attack is recently proposed in [22]. This work assumes that only one malicious agent conducts non-colluding adversarial data poisoning over the local data instances that it hosts. Our method is distinct from this work since we study consensus collaboration of multiple teachers. In addition, we investigate a more generous goal of teaching than data a pre-specified target model with that training set., which guides the learner to learn a pre-specified yet potentially malicious target model.

III. Collaborative Super Teaching

We assume K teachers and one central parameter server as the learner. Each teacher hosts a local private dataset D_i ($i \in [K]$) of size N_i. As the output of super teaching, each teacher selects a subset $S_i \subset D_i$. The learner runs the learning algorithm L on the aggregated subsets $S = \cup_{i \in [K]} S_i$ to

obtain the model. Each teacher only accesses its own data D_i during the teaching process due to the privacy-preserving regulations. Once the teaching stage terminates, the learner can further conduct federated model training to keep the local data subsets $\{S_i\}$ on the local machines, which protects teachers' data privacy after teaching. Discussing how to conduct model training is beyond our scope. Without loss of generality, we assume that the learner choses federated training.

We set the teaching goal as the value of the model parameter θ^* that the teachers expect the learner to obtain, as the setting in [6]. In the collaborative environment, the union of the selected subsets $\{S_i\}$ should be jointly helpful in inducing θ^*. Therefore we propose to define the collaborative super teaching as in (2)

$$\begin{aligned} &\hat{\theta}_S,\ b^i : i \in [K] = \underset{\hat{\theta}_S, b^i : i\in[K]}{\arg\min}\ R^*(\hat{\theta}_S) \\ s.t.\ \ &\hat{\theta}_S = \arg\min_{\theta} \sum_{i=1}^{K}\sum_{j=1}^{N_i} b_j^i \ell(\theta, x_j^i, y_j^i) + \frac{\lambda}{2}\Omega(\theta), \\ &b^i \in \{0,1\}^{N_i}, \forall i \in [K], \end{aligned} \tag{2}$$

where $R^*(\hat{\theta}_S) = \|\hat{\theta}_S - \theta^*\|$ measures the teaching risk as Euclidean distance between $\hat{\theta}_S$ and θ^*, (x_j^i, y_j^i) is the jth data instances of D_i hosted by the teacher i, and b^i is an N_i-dimensional binary-valued vector with $b_j^i = 1$ denoting the instance (x_j^i, y_j^i) is selected and $b_j^i = 0$ otherwise. ℓ is the learning loss function, $\Omega(\theta)$ is the regularization over the model complexity of the learner and λ is the regularization weight. Intuitively, there is a primitive solution to the proposed distributed teaching problem: **oblivious teaching**, where each teacher independently selects its own teaching set without collaborating with the other teachers. The independently selected subsets are aggregated to form the training set of the learner. The questions of interest are thus i) whether the oblivious teaching can reduce the teaching risk. and ii) whether it is possible to improve teaching performance by organising appropriate collaboration between the teachers, compared to the oblivious teaching.

For simplicity of analysis, we assume that $\Omega(\theta)$ is the l_2-norm penalty $\|\theta\|^2$ and the loss ℓ is a convex and τ-smooth function of θ, which holds in many cases such as the logistic loss or squared loss with bounded input space. Thus, the learning algorithm L of the learner takes the form of a convex optimization. Based on such assumption, we provide an initial answer to the question i) in Theorem 1:

Theorem 1. *Assume the model space Θ is bounded, i.e., $\forall\theta \in \Theta$, $\|\theta\| \le B$. Also assume the convex learning loss ℓ is τ-smooth, i.e., $\|\nabla\ell(\theta) - \nabla\ell(\theta')\| \le \tau\|\theta - \theta'\|, \forall\theta, \theta'$. The teaching risk is defined as $R^*(\hat{\theta}) = \frac{1}{2}\|\hat{\theta} - \theta^*\|^2$, where θ^* is the target model. Each teacher independently solves* (2) *on its own dataset D_i with regularization weight $\frac{\lambda}{K}$, and let $S = \cup_{i\in[K]} S_i$ be the aggregated dataset. Then*

$$R^*(\hat{\theta}_S) \le (\frac{\tau}{\lambda K} + \frac{1}{K^2})\sum_{i=1}^{K} R^*(\hat{\theta}_{S_i}). \tag{3}$$

Proof. Let the $(\hat{x}_j^i, \hat{y}_j^i)$ be the jth point in the selected S_i. Define $g_i(\theta) = \sum_{j=1}^{|S_i|} \ell(\theta, \hat{x}_j^i, \hat{y}_j^i) + \frac{\lambda}{2K}\|\theta\|^2$, where $|S_i|$ is the number of items in S_i. Then $\hat{\theta}_{S_i} = \arg\min_{\theta\in\Theta} g_i(\theta)$. Define

$$g(\theta) = \frac{1}{K}\sum_{i=1}^{K} g_i(\theta) = \frac{1}{K}(\sum_{i=1}^{K}\sum_{j=1}^{|S_i|} \ell(\theta, \hat{x}_j^i, \hat{y}_j^i) + \frac{\lambda}{2}\|\theta\|^2). \tag{4}$$

Then $\hat{\theta}_S = \arg\min_{\theta\in\Theta} g(\theta)$. Since $g(\theta)$ is λ-strongly convex,

$$g(\theta^*) - g(\hat{\theta}_S) \ge \frac{\lambda}{2}\|\theta^* - \hat{\theta}_S\|^2 = \lambda R^*(\hat{\theta}_S). \tag{5}$$

Thus $R^*(\hat{\theta}_S) \le \frac{1}{\lambda}(g(\theta^*) - g(\hat{\theta}_S))$. Next we upper bound $g(\theta^*) - g(\hat{\theta}_S)$. Note that $g(\theta^*) - g(\hat{\theta}_S) =$

$$\frac{1}{K}\sum_{i=1}^{K}(g_i(\theta^*) - g_i(\hat{\theta}_S)) \le \frac{1}{K}\sum_{i=1}^{K}(g_i(\theta^*) - g_i(\hat{\theta}_{S_i})), \tag{6}$$

where the last inequality is due to $\hat{\theta}_{S_i} = \arg\min_{\theta\in\Theta} g_i(\theta)$. Since the loss ℓ is τ-smooth, one can easily show that $g_i(\theta)$ is $(\tau + \frac{\lambda}{K})$-smooth, thus we have the following upper bound

$$g_i(\theta^*) - g_i(\hat{\theta}_{S_i}) \le \frac{1}{2}(\tau + \frac{\lambda}{K})\|\theta^* - \hat{\theta}_{S_i}\|^2 = (\tau + \frac{\lambda}{K})R^*(\hat{\theta}_{S_i}). \tag{7}$$

By (6) and (7), we have

$$R^*(\hat{\theta}_S) \le \frac{1}{\lambda}(g(\theta^*) - g(\hat{\theta}_S)) \le (\frac{\tau}{\lambda K} + \frac{1}{K^2})\sum_{i=1}^{K} R^*(\hat{\theta}_{S_i}). \tag{8}$$

□

Theorem 1 implies that if $\forall i$, $R^*(\hat{\theta}_{S_i}) \le c$, then $R^*(\hat{\theta}_S) \le (\frac{\tau}{\lambda} + \frac{1}{K})c$, which is a constant change. Thus by simply aggregating the selected subsets, one can achieve comparable teaching performance as each individual teacher does. However, the joint teaching performance is never guaranteed to be better than individual teaching. Therefore it is obvious that in order to achieve better joint teaching performance, the teachers should share information with each other and decide how to tweak their own teaching subset based on the selection made by the peers. This is also intuitive in real-world education, where human teachers collaboratively teach student better via communication with each other. Motivated by this observation, we define a collaborative teaching strategy that encourages information sharing between teachers to jointly minimize the teaching risk, while keeping the local data private.

A. Regularized Dual Learning for Collaborative Teaching

The dual objective of the learning paradigm for the learner gives:

$$\alpha^* = \arg\min_{\alpha} \sum_{i=1}^{K}\sum_{j=1}^{N_i} \ell^*(-\alpha_j^i) + \frac{\lambda}{2}\|Z\alpha\|^2 \tag{9}$$

where ℓ^* is the Fenchel dual of the loss function ℓ. Let $N = \sum_{i=1}^{K} N_i$ denote the number of training instances delivered by the teachers. $Z \in R^{d*N}$ denotes aggregated data matrix with each column corresponding to a data instance. The duality comes with the mapping from dual to primal variable:

$\omega(\alpha) = Z\alpha$ as given by the KKT optimality condition. α is the N-dimensional dual variable, where each α_j^i denotes the dual variable corresponding to the jth data instance hosted by teacher i. If α_j^i diminishes, the corresponding data instance Z_j^i consequently has no impact over the dual objective in (9). Thus, only the data instances with non-zero α_j^i dominates the training process. Motivated by this observation, we propose to optimize the dual objective and enforce sparsity structure of α simultaneously to achieve selection of the informative training samples in (10). Bearing in mind the goal of the collaborative teaching, we also introduce an additional quadratic penalty shrinking the gap between θ^* and the learnt model $Z\alpha$.

$$\begin{aligned}\alpha = \underset{\alpha_j^i, i\in[K]}{\arg\min} & \frac{1}{n}\sum_{i=1}^{K}\sum_{j=1}^{N_i}\ell^*(-\alpha_j^i) + \frac{\lambda}{2}\|Z\alpha\|^2 \\ & + \lambda_\theta\|\theta^* - Z\alpha\|^2 + \lambda_\alpha\sum_{i=1}^{K}\sum_{j=1}^{N_i} w_j^i|\alpha_j^i|\end{aligned} \tag{10}$$

where λ_α and λ_θ are the weight coefficients of the adaptive l_1-norm based penalization enforcing sparsity of α and the quadratic penalty minimizing the teaching risk R^*. w_j^i is data-dependent per-variable weight assigned to each dual variable α_j^i. Based on [23], w_j^i can be set up as $1/|\hat{\alpha}_j^i|$. $1/|\hat{\alpha}_j^i|$ denotes a warm-start estimate of α_j^i, which can be derived by simply calculating the Ordinary-Least-Squares solution to $\|\theta^* - Z\alpha\|^2$. The teaching objective given in (10) is apparently convex according to the property of Legendre-Fenchel transform. Thus solving (10) with gradient descent guarantees fact convergence. As enforced by the l_1-norm regularization over α, the non-zero entries of the optimal α of the objective function in (10) correspond to the most useful data instances for the learner to reach the expected teaching goal and minimize the teaching risk R^*. In practice, the learned α has a small fraction of entries with dominant magnitudes, and rest are negligible. We thus rank the data instances Z_j^i according to the magnitude of $|\alpha_j^i|$. The top-ranked $|S|$ data instances with the largest $|\alpha_j^i|$ are selected to form the final training subset for the learner. Since the selected data instances are distributed over different teachers. Solving (10) helps to jointly identify which data instances on each of the K teachers should be used to teach the learner. In the consensus optimization, each teacher learns to conduct the selection based on the decisions of the other teachers. Compared to heuristically tuning each teacher's decision, solving (10) explicitly coordinates different teachers to deliver collaborative subset selection to minimize the teaching risk globally. Furthermore, we observe that the solution to (10) enjoys a similar property as the Oracle property of adaptive Lasso [23] with an appropriately chosen λ_θ and λ_α, as given in Observation 1.

Observation 1. *Given a training set $\{(x_i, y_i)\}_{i\in[N]}$, where $x_i \in \mathcal{R}^d$. We assume that the goal of the super teaching satisfies $\theta^* = \sum_{i\in A}\alpha_i^* x_i$, where $A \subset [N]$, and α^* denotes the dual variable. We further assume that $\lambda_\theta \gg \varphi$ where φ is the empirical upper limit for the learner's classification loss on the training set. Given $\gamma > 0$, if $\frac{\lambda_\alpha}{\lambda_\theta} d^{-\frac{1}{2}} \to 0$ and $\frac{\lambda_\alpha}{\lambda_\theta} d^{-\frac{1}{2}+\frac{\gamma}{2}} \to \infty$ (see Theorem 2 in [23]), then the global optimal solution α to* (10) *must satisfy the Oracle property:* $\lim_{d\to\infty} P(\alpha = \alpha^*) = 1$.

Data: $\{z_j^i \; i = 1, 2, 3..., K, j = 1, 2, 3, ..., N_i\}$ hosted by K teachers
Input: $T \geq 1$ as the maximum iteration steps, scaling parameter $1 \leq \beta_i \leq K$, by default $\beta_i = 1$
Output: $\alpha_j^i, i = 1, 2, ..., K, j = 1, 2, ..., N_i$
Initialize: $\alpha_j^i = 0$ for all machines and $\widetilde{\theta}^{(0)} = 0$
for $t = 1$ **to** T **do**
  for *all teachers* $i = 1, 2, 3, ..., K$ *in parallel* **do**
    $\Delta\alpha^i = \underset{\Delta\alpha^i}{\arg\min}\frac{\lambda}{2}\|\widetilde{\theta}^{(t-1)} + \frac{1}{\lambda}\sum_{j=1}^{N_i}\Delta\alpha_j^i x_j^i y_j^i\|^2 + \ell^*(-\alpha^{(t-1),i} - \Delta\alpha^i) + \sum_{j=1}^{N_i} w_j^i|\alpha_j^i + \Delta\alpha_j^i| + \lambda_\theta\|\widetilde{\theta}^{t-1} + \frac{1}{\lambda}\sum_{j=1}^{N_i}\Delta\alpha_j^i x_j^i y_j^i - \theta^*\|^2$
    $\alpha^{t,i} = \alpha^{t-1,i} + \frac{\beta_i}{K}\Delta\alpha^i$
  end
  Reduce on the central parameter server
    $\widetilde{\theta}^t = \widetilde{\theta}^{t-1} + \frac{1}{\lambda}\sum_{i=1}^{K}\sum_{j=1}^{N_i}\alpha_j^{t,i} x_j^i$
  Broadcast $\widetilde{\theta}^t$ to all K teachers
end

Algorithm 1: Block-Coordinate Descent for Collaborative Super Teaching

B. privacy-preserving teaching via block-coordinate descent

We propose to use block-coordinate descent to solve (10). In each round of the descent process, we minimize (10) with respect to all α_j^i belonging to the same teacher i, while fixing all the other α as constants. The pseudo codes of the optimization procedure is given in algorithm 1.

We use $\alpha^{t,i}$ to denote the disjoint block $\{\alpha_j^i\}, j = 1, 2, 3.., N_i$ corresponding to the data instances hosted by teacher i, which are estimated at the t-th iteration. Z^i denotes the columns in the data matrix Z storing the data instances of the teacher i. In each round of iteration, we update the dual variable α^i for each of the K teachers in parallel. We assume an incremental update $\Delta\alpha^i$ based on the value of $\alpha^{t-1,i}$. This incremental variation indicates the descent direction minimizing the teaching loss with respect to the block α^i. It is estimated by minimizing the local approximation to (10), where α^i is represented as the additional combination $\alpha^{t-1,i} + \Delta\alpha^i$. β_i is the learning rate adjusting the descent step length for the block α^i. Note updating each block α^i does not require knowledge of the values for the other blocks. All the local updates need is the local dual variable value $\alpha^{t-1,i}$ obtained from the last round and the global aggregated variable $\widetilde{\theta}$ broadcasted from the central server. As such, update of each block can be conducted in parallel without inter-teacher communication. Similarly, aggregating to derive the global variable $\widetilde{\theta}$ is also a parallel process. The teachers forward the local aggregation $\sum_{j=1}^{N_i}\alpha_j^{t,i} x_j^i$ to the central server, where simply summing up the local aggregation gives the global variable value. It is worth noting that we use $\widetilde{\theta}$ to denote the global aggregation variable. It does not

imply the primal-dual correspondence, as we are solving a different problem from (10). Throughout the block-coordinate descent process, it is easy to find that i) private data hosted by any teacher is kept on its own machine in the collaboration stage. In other words, no training data is transferred directly between teachers. Furthermore, updating $\widetilde{\theta}$ only needs to transfer the local aggregation $\sum_{j=1}^{N_i} \alpha_j^{t,i} x_j^i$ to the central sever. It is difficult to infer any statistical profiles about the local data of the teachers based on solely on the local aggregation $\sum_{j=1}^{N_i} \alpha_j^{t,i} x_j^i$, which reduces the risk of unveiling local private data of one teacher to the others in the collaboration step. ii) sharing information between different teachers is conducted in the proposed method by updating the global aggregation variable $\widetilde{\theta}$ and then broadcasting the updated value to all K teachers. Communication for teaching collaboration is thus efficient, with the cost of $O(Kd)$ in each round of iteration. Moreover, according to [21], updating α^i of local teachers can be triggered with asynchronous parallelism, which allows to organize efficient teaching collaboration with large number of teachers and tight communication budget.

We demonstrate how to apply the proposed super teaching method to two prevalent learners, l_2-regularized Logistic Regression (LR) and Ridge Regression (RR).

1) Collaborative Teaching for l_2-regularized Logistic Regression: (x_j^i, y_j^i), $i = 1, 2, 3, .., K, j = 1, 2, 3, .., N_i$ denote the features and labels of the data instances hosted by all K teachers. To instantiate (10) to l_2-regularized Logistic Regression, we concretize the definition of ℓ^* with slight modification on the weight parameters, which gives:

$$\begin{aligned} \alpha = \arg\min_{\alpha} & \frac{\lambda}{2} \sum_{i=1}^{K} \sum_{j=1}^{N_i} \| \frac{1}{\lambda} \alpha_j^i x_j^i y_j^i \|^2 + \sum_{i=1}^{K} \sum_{j=1}^{N_i} \ell^*(-\alpha_j^i) \\ & + \lambda_\alpha \sum_{i=1}^{K} \sum_{j=1}^{N_i} w_j^i |\alpha_j^i| + \lambda_\theta \| \theta^* - \frac{1}{\lambda n} \sum_{i=1}^{K} \sum_{j=1}^{N_i} \alpha_j^i y_j^i x_j^i \|^2 \\ & \text{s.t} 0 \leq \alpha_j^i \leq 1 \end{aligned} \tag{11}$$

where y_j^i is the binary class label of the data instance, valued as +/-1 and $\ell^*(-\alpha_j^i) = \alpha_i \log \alpha_i + (1 - \alpha_i) \log(1 - \alpha_i)$

The collaborative super teaching for LR is defined as a box-constrained convex quadratic programming problem following the principle of algorithm 1. The optimization process is given in algorithm 2: Π is the projection operator to make the updated value of $\alpha^{(t),[k]}$ satisfy the box constraint.

2) Collaborative Teaching for Ridge Regression: Given the feature x_j^i and regression target y_j^i of each data instance, we can define the objective of collaborative teaching for Ridge Regression similarly.

$$\begin{aligned} \alpha = \arg\min_{\alpha} & \frac{1}{2} \sum_{i=1}^{K} \sum_{j=1}^{N_i} \ell^*(-\alpha_j^i) + \frac{1}{2\lambda} \| \sum_{i=1}^{K} \sum_{j=1}^{N_i} x_j^i \alpha_j^i \|^2 \\ & + \lambda_\alpha \sum_{i=1}^{K} \sum_{j=1}^{N_i} w_j^i |\alpha_j^i| + \lambda_\theta \| \theta^* - \frac{1}{\lambda} \sum_{i=1}^{K} \sum_{j=1}^{N_i} x_j^i \alpha_j^i \|^2 \end{aligned} \tag{12}$$

where $\ell^*(-\alpha_j^i) = \frac{1}{2} \|\alpha_j^i\|^2 - \alpha_j^i y_j^i$. It is thus easy to define collaboratively teaching ridge regression in algorithm 3

Initialize: $\alpha_j^i = 0$ for all teachers and $\widetilde{\theta}^{(0)} = 0$
for $t = 1$ **to** T **do**
  for *all teachers* $i = 1, 2, 3, ..., K$ *in parallel* **do**
    $\Delta\alpha^i = \arg\min_{\Delta\alpha^i} \sum_{j=1}^{N_i} \ell^*(-\alpha_j^{t-1,i} - \Delta\alpha_j^i) + \frac{\lambda}{2} \|\widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{j=1}^{N_i} \Delta\alpha_j^i x_j^i y_j^i\|^2 + \lambda_\alpha \sum_{j=1}^{N_i} w_j^i |\alpha_j^{t-1,i} + \Delta\alpha_j^i| + \lambda_\theta \|\widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{j=1}^{N_i} \Delta\alpha_j^i x_j^i y_j^i - \theta^*\|^2$
    $\alpha^{t,[k]} = \Pi(\alpha^{t-1,[k]} + \frac{\beta_K}{K} \Delta\alpha^{[k]})$
  end
  Reduce on the central parameter server
    $\widetilde{\theta}^t = \widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{i=1}^{K} \sum_{j=1}^{N_i} \alpha_j^{t,i} x_j^i$
  Broadcast $\widetilde{\theta}^t$ *to all* K *teachers*
end

Algorithm 2: Block-Coordinate Descent for Collaborative Super Teaching of l_2-Regularized Logistic Regression

Initialize: $\alpha_j^i = 0$ for teachers and $\widetilde{\theta}^0 = 0$
for $t = 1$ **to** T **do**
  for *all teachers* $i = 1, 2, 3, ..., K$ *in parallel* **do**
    $\Delta\alpha^i = \arg\min_{\Delta\alpha^i} \sum_{j=1}^{N_i} \ell^*(-\alpha_j^{t-1,i} - \Delta\alpha_j^i) + \frac{\lambda}{2} \|\widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{j=1}^{N_i} x_j^i \Delta\alpha_j^i\|^2 + \lambda_\alpha \sum_{j=1}^{N_i} w_j^i |\alpha_j^{t-1,i} + \Delta\alpha_j^i| + \lambda_\theta \|\widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{j=1}^{N_i} \Delta\alpha_j^i x_j^i - \theta^*\|^2$
    $\alpha^{t,i} = \alpha^{t-1,i} + \frac{\beta_i}{K} \Delta\alpha^i$
  end
  Reduce on the central parameter server
    $\widetilde{\theta}^t = \widetilde{\theta}^{t-1} + \frac{1}{\lambda} \sum_{i=1}^{K} \sum_{j=1}^{N_i} \alpha_j^{t,i} x_j^i$
  Broadcast $\widetilde{\theta}^t$ *to all* K *teachers*
end

Algorithm 3: Block-Coordinate Descent for Collaborative Super Teaching of Ridge Regression

C. Computational complexity and communication cost

As shown in algorithm 2 and algorithm 3, estimating the incremental update of each block α^i is a convex quadratic programming problem. With appropriately set λ_θ and λ_α, the quadratic programming problem is well scaled and can be solved in a well scalable way using polynomial time interior point algorithms, such as [24]. According to algorithm 1, only the step of aggregating the global variable $\widetilde{\theta}$ needs communication between the K teachers and the learner. Assuming that in total T iterations are needed in the block-coordinate descent in algorithm 1, the overall communication cost of running the collaborative teaching paradigm for both models is $O(TKd)$. In practices, $T = 100$ is enough to achieve convergence of the block coordinate descent.

IV. Experimental Study

A. Experimental setup

We test the proposed collaborative teaching method with both synthetic data set and real-world benchmark datasets (summarized in Table.I). For the synthetic classification and

TABLE I
SUMMARY OF PUBLIC REAL-WORLD BENCHMARK DATASETS.

Dataset	No. of Instances	No. of Features
Higgs	1,000,000	28
Superconduct	21,263	81

regression data set, we create clusters of random data instances following normal distribution. In the classification dataset, equal number of clusters are assigned to positive and negative classes to construct a balanced labelled data set. In the regression dataset, the regression target Y is given by applying random linear regressor to X. The dimensionality of each data instance is fixed to 10 universally. In the experimental study, we assume that each of the K teachers hosts $\lfloor \frac{N}{K} \rfloor$ data instances as the local data set. To generate *i.i.d.* data instances, the mean and variance of the normal distribution for data generation are kept the same for different teachers. The summary of the real-world datasets is shown by Table.I, which are used to evaluate practical performances of the proposed method over large-scale real-world data samples. The empirical study over the real-world data samples uniformly the data set and assign $\lfloor \frac{N}{K} \rfloor$ instances to each teacher.

To generate the target of teaching in the study, we run standard LR and RR on all the data samples hosted by K teachers to derive true model parameter θ^{gt}. The teaching target θ^* is given by adding a white Gaussian noise $\tau \in R^d$, as $\theta^* = \theta^{gt} + \tau$. We fix the magnitude of τ as the same of that of θ^{gt} in the following experiments to generate a teaching target with reasonable difficulty. To measure the teaching performances, we use the teaching risk R^* as the major metric. In addition, in the binary classification scenario, we apply both the teaching target θ^* and the learned parameter $\hat{\theta}_S$ on the whole data set. We count the fraction of the data instances where the output labels of the teaching target and the learned model are consistent. The higher the fraction value is, the better the teaching performance is, as the goal of teaching is to approximate the target model as close as possible. Similarly for regression, we measure r-square score between the regression output of the teaching target model and the learned model on the whole synthetic regression data set as the metric measuring the teaching quality for regression. The two additional metrics are noted as ρ_{lr} and ρ_{rr} in the experiments.

We compare the proposed collaborative teaching method to the primitive oblivious teaching strategy. To organize the oblivious teaching, we further require that each teacher selects $\frac{|S|}{K}$ instances as the identified local subset, as there is no heuristic preference over any specific teacher. The oblivious teaching is conducted by running the MINLP based teaching paradigm [6] on each teacher. The selected data instances are aggregated to form the training set of the learner. The proposed collaborative teaching method is implemented with Spark *TFOCS* library on a AWS EC2 public cloud server, with one core per teacher. For implementing the oblivious teaching method, it is difficult to find an open-sourced *MINLP* library tailored for parallel computing environments. We thus use Spark to call the MINLP solver of *NEOS* [25] for each teacher and aggregate the selected data instances to form the learner's training set. We record the running time to evaluate and compare the scalability of both teaching methods, as indicated by κ in the empirical study.

B. Benchmark with synthetic classification and regression datasets

For the tests of both classification and regression scenarios, we vary the total number N of synthetic data instances as 5000, 10000, 50000, 100000 and 500000 to cover intermediate and large-scale data volumes. For each choice of N, we further set the number of teachers K to be 5 and 10 respectively. For a fixed combination of N and K, we run 10 trials. In each trial, we draw randomly an *iid* synthetic instances and apply the proposed method of collaborative super teaching. We show the fraction of the selected subset $|S|/N$ that achieves the minimum average teaching risk of the 10 trials in Table.II and Table.III. λ_α and λ_θ are the parameters of the proposed collaborative teaching method. In the experimental study of both the classification and regression scenario, both parameters are tuned empirically using validation data instances that are generated independently besides from the benchmark set. It is interesting to find out that the values of λ_α and λ_θ are insensitive to the varying N and K. Therefore, we fix λ_α as 0.1 and λ_θ as 1000 in the binary classification scenario. In the regression scenario, they are fixed as 1 and 2000 respectively. We also run the MINLP based teaching paradigm as a centralized teacher over all the training data instances, as indicated by *MINLP* in both tables. We compare to the centralized teaching to highlight the computing efficiency of the proposed teaching paradigm.

The collaborative super teaching method selects less than 0.1 and 0.4 of the data instances to achieve accurate teaching result in both the classification and the regression test. Given N fixed, increasing K barely changes the teaching risk and the decision consistency between the target model and the model learned with the selected subset. However, it slightly increases the running time due to the increased communication cost during the global aggregation and broadcasting of $\widetilde{\theta}^t$. In all of the tests, the collaborative super teaching method runs for 85 to 150 iterations to reach convergences. With the same N, more teachers (larger K) requires more iterations before convergence. On one hand, collaborating with more teachers leads to smaller block size of the block coordinate descent, which causes slower convergence [26]. On the other hand, more teachers help to reduce the computational cost on each teacher. Depending on the computational resource budget of the teachers, we can benefit from the balance to organize efficient collaboration of the teachers. In general, compared to the oblivious teaching and the centralized teaching method, the collaborative teaching method provides significantly lower or similar teaching risk and better approximate the target model with the selected training subset in both tests. It requires distinctively less running time. The collaborative teaching

TABLE II
COMPARISON OF THE TEACHING PERFORMANCE IN THE BINARY CLASSIFICATION SCENARIO

N	**K**	**Collaborative Super Teaching**				Oblivious Super Teaching				MINLP			
		$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ	$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ	$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ
5000	5	**3.7e-2**	**0.43**	**0.95**	**7.15s**	6.0e-2	0.54	0.90	200.53s	4.0e-2	0.30	0.97	175.91s
	10	**4.5e-2**	**0.37**	**0.92**	**8.13s**	5.8e-2	0.62	0.87	197.33s				
10000	5	**1.5e-2**	**0.23**	**0.94**	**16.90s**	2.0e-2	0.63	0.87	320.69s	N/A	N/A	N/A	N/A
	10	**1.0e-2**	**0.26**	**0.97**	**18.85s**	2.0e-2	0.54	0.89	327.16s				
50000	5	**2.5e-2**	**0.35**	**0.93**	**67.20s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**2.4e-2**	**0.36**	**0.96**	**72.52s**	1.3e-2	0.22	0.94	2100s				
100000	5	**2.9e-2**	**0.33**	**0.95**	**180.24s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**3.0e-2**	**0.41**	**0.93**	**179.12s**	N/A	N/A	N/A	N/A				
500000	5	**6.4e-2**	**0.15**	**0.98**	**1064.15s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**7.2e-2**	**0.12**	**0.98**	**1100.75s**	N/A	N/A	N/A	N/A				

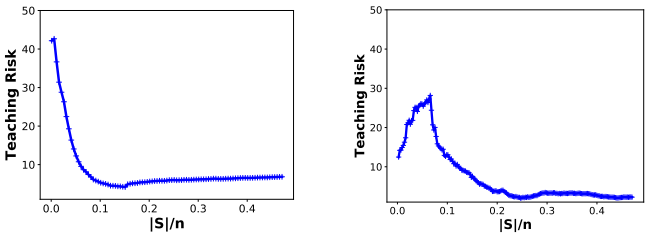
TABLE III
COMPARISON OF THE TEACHING PERFORMANCE IN THE REGRESSION SCENARIO

N	**k**	**Collaborative Super Teaching**				Oblivious Super Teaching				MINLP			
		$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ	$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ	$\|S\|/N$	$R^*(\hat{\theta}_S)$	ρ_{lr}	κ
5000	5	**1.20e-1**	**81.53**	**0.82**	**2.04s**	2.00e-1	112.93	0.71	230.53s	1.20e-1	98.09	0.76	195.84s
	10	**1.12e-1**	**80.43**	**0.83**	**2.38s**	1.80e-1	120.54	0.70	262.65s				
10000	5	**9.00e-2**	**69.52**	**0.86**	**4.25s**	2.00e-1	93.20	0.75	320.28s	2.00e-1	87.96	0.76	506.01s
	10	**7.00e-2**	**67.46**	**0.87**	**5.15s**	2.50e-1	94.20	0.76	570.32s				
50000	5	**1.00e-1**	**110.62**	**0.84**	**35.16s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**1.12e-1**	**118.36**	**0.82**	**32.53s**	N/A	N/A	N/A	N/A				
100000	5	**3.00e-1**	**99.85**	**0.88**	**81.80s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**3.00e-1**	**101.36**	**0.88**	**101.03s**	N/A	N/A	N/A	N/A				
500000	5	**3.60e-1**	**51.72**	**0.94**	**310.42s**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	10	**3.60e-1**	**49.67**	**0.95**	**395.91s**	N/A	N/A	N/A	N/A				

method costs less than 5% of the running time compared to both of the opponents according to Table.II and Table.III. Notably, the time and storage cost of the oblivious teaching becomes prohibitively expensive when $N \geq 10000$. NEOS can't get results given the large N. We write N/A in case NEOS fails to solve the MINLP problem. In contrast, the proposed collaborative teaching method can still produce accurate teaching output with acceptable time cost. Despite of the difference of implementation details of the three teaching methods, the difference of running time confirms the superior computational efficiency of the proposed collaborative teaching paradigm. Benefited from the consensus optimization process, the proposed collaborative teaching paradigm provides a highly scalable solver to the distributed super teaching task. Moreover, it is interesting to find out the centralized teaching paradigm performs better than the oblivious teaching method. This observation is consistent with what Theorem.1 unveils: information sharing is the key to achieve good teaching cooperation. Stand-alone teaching without inter-communication between teachers can do harm to the teaching performance.

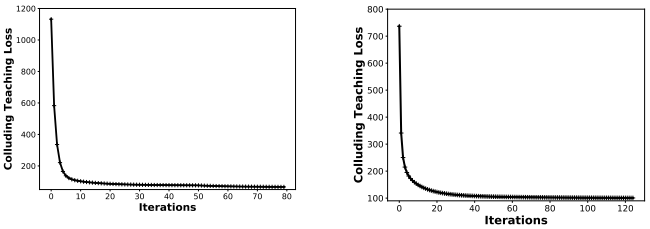
C. Benchmark with real-world data sets

Two real-world data sets, *Higgs* and *Superconduct*, are employed to test the collaborative super teaching method for l_2-regularized Logistic Regression and Ridge Regression respectively. The number of the teachers is chosen to be 5 on both data sets. The setting of λ_α, λ_θ, the computing platform and the teaching goal follow the same setting as the test on synthetic data. For Higgs data set, we randomly sample 1000000 instances from the whole set for 10 times and re-run the proposed method on the sampled Higgs data samples. Figure 1a illustrates the variation of the averaged teaching risk of the proposed collaborative teaching method by incrementally increasing the number of the jointly selected instances. In Figure 2a, we demonstrate how the objective function value of the proposed teaching method diminishes as the iterative block-coordinate descent runs. On Higgs data set, the proposed method selects only 15% of the 1000000 instances to achieve the teaching risk of 3.28. The corresponding consistency score ρ_{lr} is 0.99. It indicates that the learner manages to approximate the expected target model perfectly with the selected subsets given the teachers. Interestingly, the teaching risk declines at first as $|S|/N$ increases to 15%. After this turning point, the teaching risk begins to increase again. The observation is consistent with our intuitive understanding about the teaching process. Insufficient and too many data instances can do harm equally to the teaching performances. From Figure 2a, we can find the objective function value of the proposed collaborative teaching method declines rapidly within 50 iterations. In this experiment, the consensus optimization process of the proposed collaborative teaching paradigm converges with 80 iteration steps. which costs 2095.48s. Similarly, Figure 1b shows the declination of teaching risk by increasing gradually the number of selected instances on *Superconduct* data set.



(a) Higgs Data (b) Superconduct Data

Fig. 1. Teaching risk variation with different number of the selected data instances



(a) Higgs Data (b) Superconduct Data

Fig. 2. Convergence of the quadratic programming based collaborative teaching process

As seen in the figure, the proposed teaching method selects 25% of the data instances to reach the teaching risk of 2.23 and ρ_{rr} of 0.97. Figure 2b illustrates the declination of the objective functions values on *Superconduct* achieves. Similar pattern of the teaching risk variation is witnessed in Figure 1b, compared to Figure 1a. The turning point of the teaching risk curve confirms empirically the existence of the optimal subset for teaching. Based on the selected subset, the learner can accurately fit the target regression model. Minimizing the collaborative teaching objective for Ridge Regression converges within 125 iterations, which costs only 35.12s.

V. Conclusion and Discussion

In this paper, we explore how to organize scalable, collaborative, and privacy-preserving super teaching with multiple teachers. We formulate a distributed convex optimization problem for conducting consensus super teaching with varying number of teachers, and adopt a block descent based solver to optimize each teacher's selection on teaching items. Our approach preserves data privacy during the collaborative teaching process. We show that the proposed collaborative teaching scheme can achieve lower teaching risk than the non-collaborative scheme. Empirical results on both synthetic and real-word data sets confirm the superior performance of the proposed collaborative teaching method over the non-collaborative solution. Future work will study practical use of distributed and privacy-preserving super teaching based on the proposed collaborative teaching framework, e.g., we plan to explore the teaching goals that are realistic to practical use, such as AUC-maximization oriented goals.

References

[1] S. A. Goldman and M. J. Kearns, "On the complexity of teaching," *Journal of Computer and System Sciences*, vol. 50, no. 1, pp. 20–31, 1995.

[2] A. Shinohara and S. Miyano, "Teachability in computational learning," *New Generation Computing*, vol. 8, no. 4, pp. 337–347, 1991.

[3] S. Zilles, S. Lange, R. Holte, and M. Zinkevich, "Models of cooperative teaching and learning," *Journal of Machine Learning Research*, vol. 12, no. Feb, pp. 349–384, 2011.

[4] J. Zhu, "Machine teaching for bayesian learners in the exponential family," in *Advances in Neural Information Processing Systems*, 2013, pp. 1905–1913.

[5] Y. Chen, A. Singla, O. Mac Aodha, P. Perona, and Y. Yue, "Understanding the role of adaptivity in machine teaching: The case of version space learners," *arXiv preprint arXiv:1802.05190*, 2018.

[6] Y. Ma, R. Nowak, P. Rigollet, X. Zhang, and X. Zhu, "Teacher improves learning by selecting a training subset," in *International Conference on Artificial Intelligence and Statistics*, 2018, pp. 1366–1375.

[7] L. Liberti, "Undecidability and hardness in minlp," *RAIRO Operations Research*, to appear.

[8] E. Karabulut, "Distributed integer programming," *Phd Thesis, GeorgiaTech*, 2017.

[9] J. Liu, X. Zhu, and H. Ohannessian, "The teaching dimension of linear learners," in *International Conference on Machine Learning*, 2016, pp. 117–126.

[10] F. J. Balbach, "Measuring teachability using variants of the teaching dimension," *Theoretical Computer Science*, vol. 397, no. 1-3, pp. 94–113, 2008.

[11] T. Doliwa, G. Fan, H. U. Simon, and S. Zilles, "Recursive teaching dimension, vc-dimension and sample compression," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3107–3131, 2014.

[12] L. Haug, S. Tschiatschek, and A. Singla, "Teaching inverse reinforcement learners via features and demonstrations," in *Advances in Neural Information Processing Systems*, 2018, pp. 8472–8481.

[13] W. Liu, B. Dai, X. Li, Z. Liu, J. M. Rehg, and L. Song, "Towards black-box iterative machine teaching," *arXiv preprint arXiv:1710.07742*, 2017.

[14] W. Liu, B. Dai, A. Humayun, C. Tay, C. Yu, L. B. Smith, J. M. Rehg, and L. Song, "Iterative machine teaching," in *International Conference on Machine Learning*, 2017, pp. 2149–2158.

[15] K. R. Patil, J. Zhu, Ł. Kopeć, and B. C. Love, "Optimal teaching for limited-capacity human learners," in *Advances in neural information processing systems*, 2014, pp. 2465–2473.

[16] S. Mei and X. Zhu, "Using machine teaching to identify optimal training-set attacks on machine learners." in *AAAI*, 2015, pp. 2871–2877.

[17] S. Alfeld, X. Zhu, and P. Barford, "Data poisoning attacks against autoregressive models." in *AAAI*, 2016, pp. 1452–1458.

[18] Y. Ma, K.-S. Jun, L. Li, and X. Zhu, "Data poisoning attacks in contextual bandits," in *International Conference on Decision and Game Theory for Security*. Springer, 2018, pp. 186–204.

[19] Y. Fan, F. Tian, T. Qin, X.-Y. Li, and T.-Y. Liu, "Learning to teach," in *International Conference on Learning Representations*, 2018.

[20] J. Konecny, H. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *NIPS Workshop on Private Multi-Party Machine Learning*, 2016.

[21] M. Jaggi, V. Smith, M. Takac, J. Terhorst, S. Krishnan, T. Hofmann, and M. I. Jordan, "Communication-efficient distributed dual coordinate ascent," in *Advances in neural information processing systems*, 2014, pp. 3068–3076.

[22] A. N. Bhahpji, S. Charkraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," in *arXiv preprint arXiv:1811.12470*, 2018.

[23] H. Zhou, "The adaptive lasso and its oracle properties," *Journal of the American Statistical Association*, pp. 1418–1429, 2006.

[24] Y. Ye and E. Tse, "An extension of karmarkar's projective algorithm for convex quadratic programming," *Mathematical Programming*, pp. 157–179, 1989.

[25] E. D. Dolan, "The neos server 4.0 administrative guide," Mathematics and Computer Science Division, Argonne National Laboratory, Technical Memorandum ANL/MCS-TM-250, 2001.

[26] A. Beck and L. Tetruashvili, "On the convergence of block coordinate descent type methods," *SIAM Journal of Optimization*, pp. 2037–2060, 2013.