Differentially Private Linear Regression over Fully Decentralized Datasets

Yang Liu
Tencent Cloud Product Department
Tencent
Shenzhen 518057, China
clarkieliu@tencent.com

Xiong Zhang
Tencent Cloud Product Department
Tencent
Shenzhen 518057, China
farleyzhang@tencent.com

Shuqin Qin
Tencent Cloud Product Department
Tencent
Shenzhen 518057, China
sookieqin@tencent.com

Xiaoping Lei
Tencent Cloud Product Department
Tencent
Shenzhen 518057, China
edenlei@tencent.com

Abstract

This paper presents a differentially private algorithm for linear regression learning in a decentralized fashion. Under this algorithm, privacy budget is theoretically derived, in addition to that the solution error is shown to be bounded by $O\left(\frac{1}{t}\right)$ for $O\left(\frac{1}{\epsilon}\right)$ descent step size and $O\left(\exp\left(\frac{1}{t^e}\right)\right)$ for $O\left(\frac{1}{\epsilon}\right)$ descent step size.

1 Introduction

In recent years, optimization and learning among fully decentralized parties are drawing much attention [Nedic and Ozdaglar 2009, Nedic et al. 2010, Boyd et al. 2011]. However, privacy concerns are not taken into account in much of the work. Although Huang et al. [2015] presents a private distributed convex optimizer by incorporating the famous notion of differential privacy [Dwork 2011], too strong boundedness assumptions on the objectives must hold. In this paper, we specify the objective as the famous least squares, and provide a differentially private decentralized solver, as well as privacy and accuracy results with relaxed assumptions.

2 Problem Definition

2.1 Decentralized Datasets over Networks

Let $V = \{1, \ldots, k\}$ represent a group of decentralized parties that aim to participate in a global computational task. As a setup of this paper, the parties in $V$, termed as nodes, are peer-to-peer interconnected to locally establish two-way communication, described by edges in a set of unordered pair of nodes $E = \{\{i, j\} : i, j \text{ are connected}, i, j \in V\}$. Based on the edge set $E$, one can define the neighbor set of node $i$ as $N_i = \{j : \{i, j\} \in E\} \cup \{i\}$. Over such a network $G = (V, E)$, which is assumed to be connected throughout this paper, nodes $i \in V$ hold mutually exclusive and homogeneous datasets $D_i \in \mathbb{R}^{n_i \times m} \times \mathbb{R}^{n_i}$, respectively, including the design matrix $X_i \in \mathbb{R}^{n_i \times m}$ and the label vector $y_i \in \mathbb{R}^{n_i}$. One of the foundational assumptions of this paper is that $D_i$ is seen as privacy by each node $i$.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.
2.2 Existing Decentralized Linear Regression Algorithm

Linear regression is a common model that arises in various disciplines. Consider a design matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ and a label vector $\mathbf{y} \in \mathbb{R}^n$. Then the learning goal of linear regression is to solve the following least-squares problem:

$$\min_{\mathbf{\beta} \in \mathbb{R}^m} \frac{1}{2} ||\mathbf{X}\mathbf{\beta} - \mathbf{y}||^2.$$  \hfill (1)

It is well-known that (1) yields a unique optimal estimate $\mathbf{\beta}^* = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$ if $\mathbf{X}$ has full column rank. By letting $n = \sum_{i=1}^{k} n_i$, $\mathbf{X} = [\mathbf{X}_1^\top \ldots \mathbf{X}_k^\top]^\top$ and $\mathbf{y} = [\mathbf{y}_1^\top \ldots \mathbf{y}_k^\top]^\top$, we finally obtain a decentralized linear regression modelling task (1) over network $G$. A fully decentralized algorithm for solving (1) is described by the following dynamics [Nedic et al. 2010]:

$$\mathbf{\beta}_i(t+1) = \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{\beta}_j(t) - \alpha(t)\nabla L_i(\mathbf{\beta}_i(t)),$$  \hfill (2)

where $t = 0, 1, 2, \ldots$ is the discretized time, $\mathbf{\beta}_i(t)$ is node $i$’s current estimate towards the global model, edge weight $w_{ij} > 0$ is defined over $j \in \mathcal{N}_i$ satisfying $w_{ij} = w_{ji}$ and $\sum_{j \in \mathcal{N}_i} w_{ij} = 1$ for all $i \in \mathcal{V}$, $\alpha : \mathbb{Z}^+ \to \mathbb{R}^+$ is the step size, and $L_i(\mathbf{\beta}) = \frac{1}{2}||\mathbf{X}_i\mathbf{\beta} - \mathbf{y}_i||^2$. It was proved that if $\alpha(t) = \alpha \sum_{t=0}^{\infty} \frac{1}{(t+1)^c} > 0$, then $\lim_{t \to \infty} \mathbf{\beta}_i(t) = \mathbf{\beta}^*$ for all $i \in \mathcal{V}$ [Liu et al. 2018]. Typical selections of $\alpha(t)$ include $\alpha(t) = \frac{c}{1+d^t}$ with $c, d > 0$ and $0 < e \leq 1$. Evidently, the contents shared among nodes are $\{\mathbf{\beta}_i(t)\}_{i \in \mathcal{V}, t \geq 0}$, which contain the information of $\nabla L_i$ and thereby $D_i$. When confronted with global adversaries capable of observing the communication contents, the algorithm (2) leads to undesirable privacy disclosure. Therefore, a privacy-preserving version of (2) is demanded.

3 Main Results

In this section, we propose a privacy-preserving version of (2), and provide corresponding differential privacy and accuracy analysis. To facilitate the presentation of our algorithm, we first introduce the following assumption.

**Assumption 1.** All nodes of the network $G$ knows that the optimal estimate $\mathbf{\beta}^* \in \mathbb{R}^m$ falls into a compact and convex set $\Omega \subset \mathbb{R}^m$ with $D_\Omega = \sup_{\mathbf{\beta} \in \Omega} ||\mathbf{\beta}||$.

Note that Assumption 1 is reasonable in the sense that heuristic approaches can be applied to find $\Omega$. For example, if rank($\mathbf{X}_i$) = $m$, each node $i$ can present a convex set $\Omega_i \subset \mathbb{R}^m$ containing its local optimal estimate $\mathbf{\beta}_i^* = \arg\min_{\mathbf{\beta} \in \mathbb{R}^m} L_i(\mathbf{\beta})$, and $\Omega$ can be set as a convex hull of $\bigcup_{i \in \mathcal{V}} \Omega_i$. Such methods are out of scope, and thereby not comprehensively investigated in this paper.

3.1 Privacy-Preserving Algorithm

Define $D_\Omega(\mathbf{\beta}) = \inf_{\mathbf{\beta}' \in \Omega} ||\mathbf{\beta} - \mathbf{\beta}'||$ as the projection onto $\Omega$. Inspired by (2), we provide the following privacy-preserving linear regression algorithm that terminates in finite time $T \geq 1$.

**Algorithm 1 T-step Privacy-Preserving Linear Regression**

1: Set $t \leftarrow 0$ and initialize $\mathbf{\beta}_i(0)$ for all $i \in \mathcal{V}$.
2: Each node $i$ draws $\omega_i(t) \in \mathbb{R}^m$ from the distribution $Lap_r(\mathbf{\omega}(t))$ satisfying $\lim_{t \to \infty} \mathbf{\omega}(t) = 0$.
3: Each node $i$ computes and propagates $\mathbf{\beta}_i(t) \leftarrow \mathbf{\beta}_i(t) + \mathbf{\omega}(t)$ to its neighbors $j \in \mathcal{N}_i$.
4: Each node $i$ computes the projected state $\mathbf{\beta}_i^2(t) \leftarrow D_\Omega(\mathbf{\beta}_i^2(t))$.
5: Each node $i$ updates its state by $\mathbf{\beta}_i(t+1) \leftarrow \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{\beta}_j^2(t) - \alpha(t)\nabla L_i(\mathbf{\beta}_i^2(t))$.
6: Set $t \leftarrow t + 1$. Algorithm terminates if $t = T$, otherwise go to Step 2.
As can be noted, under Algorithm 1 each node injects Laplace random noise before true estimate for any $i$. For Algorithm 1, we provide the following theorem.

**Theorem 1.** Consider two network datasets $D = (X, y)$ and $D' = (X', y')$ in $\mathbb{R}^{n \times m} \times \mathbb{R}^n$ with $n = \sum_{i=1}^k n_i$. Then $D$ and $D'$ are said to be $(\delta_X, \delta_y)$-adjacent if there exists $i \in \{1, \ldots, k\}$ such that (i) $\|X_i\|, \|X'_i\| \leq \delta_X$ and $\|y_i\|, \|y'_i\| \leq \delta_y$; (ii) $X_j = X'_j$ and $y_j = y'_j$ for all $j \neq i$.

Clearly, the adversaries against Algorithm 1 observe all communication contents among nodes $\{\beta_i(t)\}_{i \in V, t=0,\ldots,T-1}$, based on which they aim to infer the privacy $D$. Such an adversarial relation can be intrinsically described by a mapping $M_T : \mathbb{R}^{n \times m} \times \mathbb{R}^n \times \mathbb{R}^{km} \to \mathbb{R}^{kmT}$ with

$$M_T(D, \{\beta_i(0)\}_{i \in V}) = \{\beta_i(t)\}_{i \in V, t=0,\ldots,T-1}.$$  

Then the following definition is provided on the differential privacy of Algorithm 1.

**Definition 2.** Algorithm 1 in $T$-step preserves $\epsilon$-differential privacy under $(\delta_X, \delta_y)$-adjacency if for all $R \subset \mathbb{R}^{kmT}$ and for all $\{\beta_i(0)\}_{i \in V}$ in $\mathbb{R}^{km}$, there holds

$$\Pr(M_T(D, \{\beta_i(0)\}_{i \in V}) \in R) \leq e^\epsilon \Pr(M_T(D', \{\beta'_i(0)\}_{i \in V}) \in R)$$

for all $(\delta_X, \delta_y)$-adjacent network datasets $D, D' \in \mathbb{R}^{n \times m} \times \mathbb{R}^n$.

For Algorithm 1 we provide the following theorem.

**Theorem 1.** Let Assumption 1 hold. Then there exists finite $\epsilon > 0$ such that Algorithm 1 in $T$-step preserves $\epsilon$-differential privacy under $(\delta_X, \delta_y)$-adjacency as $T$ goes to infinity if $\{\alpha_i(t)\}_{i=0}^\infty$ is summable. In particular, if $\alpha(t) = \frac{c_n}{(t+d_n)^a}$ and $v(t) = \frac{c_v}{(t+d_v)^\nu}$ with $c_n, a_n, c_v, \nu > 0$ and $1 < d_n + 1 \leq d_n$, then Algorithm 1 in $T$-step preserves $\epsilon$-differential privacy with $n_M = \max\{n_i \colon i \in V\}$.

**Proof.** We will use the compact notation $\beta(t) = [\beta_1(t)^T \ldots \beta_k(t)^T] \in \mathbb{R}^{km}$ for $\beta_i(t)$, and the same form will also appear for $\beta_i'(t)$ and $\beta_i''(t)$, whose introduction will be omitted. The underlying dynamics of Algorithm 1 can be written as

$$\beta'(t+1) = (W \otimes \mathbf{I}_m)P_{\Omega}^{\dagger}(\beta(t)) - \alpha(t)G(P_{\Omega}^{\dagger}(\beta(t))) + \omega(t+1), \quad (3)$$

where the $ij$-th element of $W \in \mathbb{R}^{k \times k}$ equals $w_{ij}$ if $j \in N_i$ and zero otherwise, $P_{\Omega}^{\dagger}(\beta(t)) = [P_1^{\dagger}(\beta(t))^T \ldots P_k^{\dagger}(\beta(t))^T]$, and $G(P_{\Omega}^{\dagger}(\beta(t))) = [\nabla L_1(P_1^{\dagger}(\beta(t)))^T \ldots \nabla L_k(P_k^{\dagger}(\beta(t)))^T]^T - X_{\Omega}^{\dagger}(\beta(t)) - \tilde{y}$ with $X = \text{diag}(X_1, X_2, \ldots, X_k)$ and $\tilde{y} = [y_1^T, \ldots, y_k^T]^T$. Define $M(t)(D, \beta(t)) = \beta'(t+1)$ such that $M_T(\{\beta_i(0)\}_{i \in V}) = \{M^{(\tau)} \circ \cdots \circ M^{(0)} : \tau = 0, \ldots, T-1\}$ when omitting $D$. Then for any $D, D'$ differing at node $i$’s dataset w.l.o.g., there hold for all $t \geq 0$ based on (3)

$$Pr(M(t)(D, \beta(t)) = \beta'(t+1)) = \frac{\Pr(M(t)(D', \beta(t)) = \beta'(t+1))}{\Pr(M(t)(D', \beta(t)) = \beta'(t+1))} \leq \exp(\alpha(t)v^{-1}(t+1)(\|\tilde{X} - X'\|_1 + \|\tilde{y} - S\|_1))$$

where

$$Pr(M(t)(D, \beta(t)) = \beta'(t+1)) \leq \exp(\alpha(t)v^{-1}(t+1)(\|X_i^T X_i - X_i^T y_i + y_i^T y_i - X_i^T y_i)|_1 + \|X_i^T y_i - X_i^T y_i\|_1)),$$  

(4)
where a) is from the Laplace distribution and b) is an application of norm inequalities. Based on norm inequalities and equivalence Further, based on (7), we have

\[ \|X_i^T y_i^* - X_i^T y_i^\|_1 \leq 4\delta X \sqrt{mn}. \]  

(5)

Similarly, we have

\[ \|X_i^T, y_i^* - X_i^T y_i^\|_1 \leq 4\delta X \sqrt{mn}. \]  

(6)

According to (4), (5) and (6),

\[
\Pr(M^{(t)}(D, \beta^b(t)) = \beta^b(t+1)) = \Pr(M^{(t)}(D', \beta^b(t)) = \beta^b(t+1)) \leq \exp \left( 4\delta X \sqrt{mn} (\delta X B \Omega \sqrt{km} + \delta_y) a(t)v^{-1}(t+1) \right). 
\]

(7)

Based on (7) and the composition property and the composition property [McSherry 2009], this proof is completed.

3.3 Accuracy Analysis

**Theorem 2.** Let Assumption[7] hold. Suppose \( \alpha(t) = O\left(\frac{1}{t^b}\right) \) with \( 0 < e_\alpha \leq 1 \) and \( v(t) = O\left(\frac{1}{t^c}\right) \) with \( e_v > 0 \). Then under Algorithm[7] there holds

\[
\sum_{i \in V} E[\|\beta^i_0(t) - \beta^*\|] = \begin{cases} 
O(t) \quad & \text{if } e_\alpha = 1; \\
O(\exp(t^{1-e_v})) \quad & \text{otherwise.} 
\end{cases} 
\]

**Proof.** We will continue to use the notations in the proof of Theorem[11] Define \( e(t) = \beta^b(t) - \mathbf{1} \otimes \beta^* \). By subtracting \( \mathbf{1} \otimes \beta^* \) on both sides of (3), one has

\[
e(t + 1) = (W \otimes I - \alpha(t) \tilde{X}) e(t) + \alpha(t)(\tilde{y} - \tilde{X}(1 \otimes \beta^*)) + \omega(t + 1). 
\]

(8)

Then it follows (8)

\[
\|e(t + 1)\|^2 \leq e(t)^T (W \otimes I - \alpha(t) \tilde{X})^2 e(t) + \alpha^2(t) \|\tilde{y} - \tilde{X}(1 \otimes \beta^*)\|^2 + \|\omega(t + 1)\|^2 + \alpha(t)^2 \|\tilde{y} - \tilde{X}(1 \otimes \beta^*)\| \| W \otimes I - \alpha(t) \tilde{X} \| \| e(t) \| + g(\omega(t + 1)),
\]

(9)

where \( g : \mathbb{R}^{km} \to \mathbb{R}^{km} \) is linear. Due to the nonnegativity and irreducibility of \( W \) Further, based on (3), there holds \( -1 \leq \|W\| < 1 \) and thereby \( \|W \otimes I - \alpha(t) \tilde{X}\| \leq 1 + \alpha(t) \). Then by (2),

\[
E\|e(t + 1)\|^2 = O(\left(1 + \alpha(t)\right)^2 E\|e(t)\|^2 + \alpha(t)(1 + \alpha(t))\|e(t)\| + \alpha^2(t) + v^2(t))
\]

which further leads to

\[
E\|e(t + 1)\| = O\left(\left(1 + \alpha(t)\right)E\|e(t)\| + \alpha(t) + v(t)\right)
\]

\[
= O\left(\prod_{\tau=0}^t (1 + \alpha(\tau)) + \sum_{\tau=0}^t (\alpha(\tau) + v(\tau)) \prod_{\kappa=\tau+1}^t (1 + \alpha(\kappa))\right)
\]

\[
= O\left(\exp\left(\sum_{\tau=0}^t \alpha(\tau)\right) + \sum_{\tau=0}^t (\alpha(\tau) + v(\tau)) \exp\left(\sum_{\kappa=\tau+1}^t \alpha(\kappa)\right)\right). 
\]

(10)

It is a fact \( \sum_{\tau=0}^t \frac{1}{\tau} = O\left(\int_{t'}^{-1} \frac{1}{\tau} d\tau\right) \) for all \( t' \geq 1 \). Based on (10), one has

\[
E\|e(t + 1)\| = \begin{cases} 
O\left(t + t \sum_{\tau=0}^t \frac{\alpha(\tau) + v(\tau)}{\tau}\right) \quad & \text{if } e_\alpha = 1; \\
O\left(\exp(t^{1-e_v}) + \exp(t^{1-e_v}) \sum_{\tau=0}^t \frac{\alpha(\tau) + v(\tau)}{\exp(\tau^{1-e_v})}\right) \quad & \text{otherwise.} 
\end{cases} 
\]

(11)

Clearly, both \( \sum_{\tau=0}^\infty \frac{\alpha(\tau) + v(\tau)}{\tau} \) and \( \sum_{\tau=0}^\infty \frac{\alpha(\tau) + v(\tau)}{\exp(\tau^{1-e_v})} \) are convergent, the proof is completed by (11).
4 Conclusions

In this paper, a differentially private decentralized algorithm for linear regression was proposed. Not only a theoretic privacy budget was provided, but the precision was carefully investigated and shown to be bounded by $\mathcal{O}(t)$ or $\mathcal{O}(\exp(t^{1-c}))$. Future work includes the tradeoff analysis between efficiency and privacy, and the relaxation of the projection operation.

References

Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48, 2009.

Angelia Nedic, Asuman Ozdaglar, and Pablo A Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 55(4):922–938, 2010.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Zhenqi Huang, Sayan Mitra, and Nitin Vaidya. Differentially private distributed optimization. In *Proceedings of the 2015 International Conference on Distributed Computing and Networking*, page 4. ACM, 2015.

Cynthia Dwork. Differential privacy. *Encyclopedia of Cryptography and Security*, pages 338–340, 2011.

Yang Liu, Youcheng Lou, Brian Anderson, and Guodong Shi. Network flows that solve least squares for linear equations. *arXiv preprint arXiv:1808.04140*, 2018.

Roger A Horn and Charles R Johnson. *Matrix Analysis*. Cambridge university press, 2012.

Frank D McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 19–30. ACM, 2009.