Convex optimization-based Privacy-Preserving Distributed Least Squares via Subspace Perturbation

Li, Qiongxiu; Heusdens, Richard; Christensen, Mads Græsbøll

Published in:
28th European Signal Processing Conference (EUSIPCO)

DOI (link to publication from Publisher):
10.23919/Eusipco47968.2020.9287473

Publication date:
2021

Document Version
Accepted author manuscript, peer reviewed version

Convex optimization-based Privacy-Preserving Distributed Least Squares via Subspace Perturbation

Qiongxiu Li[1], Richard Heusdens[2,3], Mads Græsbøll Christensen[1]
[1]Audio Analysis Lab, CREATE, Aalborg University, Denmark, {qili, mgc}@create.aau.dk
[2]Circuits and Systems group, Delft University of Technology, The Netherlands, r.heusdens@tudelft.nl
[3]Netherlands Defence Academy, The Netherlands

***Abstract*—Over the past decades, privacy-preservation has received considerable attention, not only as a consequence of regulations such as the General Data Protection Regulation in the EU, but also from the fact that people are more concerned about data abuse as the world is becoming increasingly digitized. In this paper we propose a convex optimization-based subspace perturbation approach to solve privacy-preserving distributed least squares problems. Based on the primal-dual method of multipliers, the introduced dual variables will only converge in a subspace determined by the graph topology and do not converge in its orthogonal complement. We, therefore, propose to exploit this property for privacy-preservation by using the non-converging part of the dual variables to perturb the private data, thereby protecting it from being revealed. Moreover, we prove that the proposed approach is secure under both eavesdropping and passive adversaries. Computer simulations are conducted to demonstrate the benefits of the proposed approach through its convergence properties and accuracy.**

***Index Terms*—Distributed least squares, subspace, privacy, noise perturbation, convex optimization**

I. Introduction

In modern systems, such as smart grids and smart internet-of-things, the trend is to have collaborations between different parties. This distributed processing has a number of advantages over centralised processing, like avoiding a single point of failure and being robust against changes in the network topology. Such distributed systems usually require data exchange among the parties. These data, more often than not, contain sensitive information about individual parties/agents. For example, it was shown in [1] that even electricity consumption data can reveal sensitive information about the consumers' privacy such as whether the consumer has illnesses/disabilities or not. To address such privacy issues in distributed processing, in this paper we focus on privacy-preserving distributed least squares as it is a fundamental problem and serves as a building block to many other problems such as robust signal de-noising and linear regression in machine learning.

The privacy issue in distributed processing has been addressed in the literature by either protecting the private data using secure multiparty computation (SMPC) techniques or by perturbing it with noise insertion. SMPC [2] aims to jointly compute a function among a group of parties while keeping each party's input private. Popular SMPC protocols like secret sharing, homomorphic encryption, garbled circuits and hybrid methods have been applied in linear regression problems in machine learning [3]–[6]. However, these SMPC-based frameworks usually assume either a non-colluding trusted third party (TTP) or a small network with only a few computing parties. Consequently, they are quite far from being applied in large scale networks such as wireless sensor networks and many other applications where a TTP is hard to implement. To alleviate these problems, both distributed computation and SMPC were employed in [7] for solving the privacy-preserving recursive least squares problems. Unfortunately, it comes at the cost of high communication complexity.

Noise insertion can be an attractive alternative as it is lightweight and usually does not require a TTP. A noise insertion framework for perturbing private data by balancing the privacy level with the output accuracy (referred to as differential privacy (DP) [8]), has been applied in many applications like robust statistic [9], Kalman filtering [10] and distributed average consensus [11], etc. In principle, it can also be applied to the distributed least squares problem. However, as stated in [11], there is an inherent trade-off between privacy and accuracy, and they can not be achieved simultaneously.

To address the above mentioned limitations, we here propose a novel convex optimization-based subspace perturbation approach which protects the private data by adding noise in a particular subspace. We use the primal-dual method of multipliers (PDMM) [12], [13], a distributed algorithm for solving constrained convex optimization problems, to illustrate the main idea of subspace perturbation, but the approach will work with other algorithms, like ADMM, as well. A number of attractive properties of the proposed approach are: 1) it is fundamentally different from the DP approaches as it is able to achieve both privacy and accuracy at the same time; 2) it requires no TTP and has a low computational complexity; 3) it converges at a rate independent of the privacy level and 4)it is secure under both passive and eavesdropping adversaries.

II. Fundamentals and problem Setup

In this section, we will first recall the fundamentals of the distributed least squares and explain the motivation for privacy-preservation. Next, we introduce the so-called adversary models, an essential concept when considering privacy, and then state the problem setup.

A. Distributed least squares

Given a distributed network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, \dots, n\}$ the set of nodes and $\mathcal{E} = \{e_1, \dots, e_m\}$ the

set of edges. The neighbourhood of node i is denoted as $\mathcal{N}_i = \{j|(i,j) \in \mathcal{E}\}$ and $d_i = |\mathcal{N}_i|$. Let $B \in \mathbb{R}^{m\times n}$ denote the incidence matrix defined as $B_{li} = B_{i|j} = 1$ if and only if $e_l = (i,j) \in \mathcal{E}$ and $i < j$, $B_{li} = B_{i|j} = -1$ if and only if $e_l = (i,j) \in \mathcal{E}$ and $i > j$.

The goal of distributed least squares is to find a solution of an overdetermined system (set of equations in which there are more equations than unknowns), where each node only knows part of the equations and is only able to exchange information with its neighbours. Let $Q_i \in \mathbb{R}^{N_i\times u}$, $N_i > u$, denote a matrix containing the input observations of node i. That is, each node i has N_i observations and each observation contains an u-dimensional feature vector. Moreover, let $y_i \in \mathbb{R}^{N_i}$ denote the decision vector observed by node i. Stacking all the local information such that $Q = [Q_1^T, \ldots, Q_n^T]^T \in \mathbb{R}^{N\times u}$ and $y = [y_1^T, \ldots, y_n^T]^T \in \mathbb{R}^N$ where $N = \sum_{i\in\mathcal{V}} N_i$, the least-squares problem is given by

$$\min_x \frac{1}{2}\|y - Qx\|_2^2.$$

We can formulate the least-squares problem as a distributed linearly-constrained convex optimization problem given by

$$\begin{aligned} \min_{\{x_i\}} f(x) &= \sum_{i\in\mathcal{V}} \frac{1}{2}\|y_i - Q_i x_i\|_2^2 \\ \text{s.t. } & x_i - x_j = 0, \forall (i,j) \in \mathcal{E}, \end{aligned} \tag{1}$$

where $x_i \in \mathbb{R}^u$ denotes the local estimated least-squares solution at node i. A number of distributed optimizers (e.g., ADMM, PDMM) has been proposed to solve the above problem by only exchanging information in the local neighbourhood. At every iteration k, each node i updates its local estimate $x_i^{(k)}$ based on a certain local updating function and then sends it to its neighbours. Generally, this local updating function requires local information of node i (that is, Q_i, y_i) to guarantee that $x_i^{(k)}$ converges to the global optimum solution $x^* = \arg\min_x \frac{1}{2}\|y - Qx\|_2^2$.

B. Privacy concerns

The local information (input observations Q_i and decision vector y_i) of each node is considered as private data and should be protected from being revealed. This is because it usually contains sensitive private information about individuals. For example, assume a number of hospitals participate in a research project with the aim of obtaining a predictive model by collaboratively learning all the data in their medical data sets. However, releasing this medical data violates the privacy regulation as it contains sensitive information of the patients such as their health conditions and insurance records. As mentioned earlier, at each iteration of the distributed computation, each node will send out the updated $x_i^{(k)}$ where the related updating function usually takes the private data Q_i and y_i as inputs. As a consequence, the updated $x_i^{(k)}$ carries information about the concerned private data and thus revealing it will inevitably cause loss of privacy. Such privacy issues will be investigated and addressed in the rest of the paper.

C. Adversary model

The adversary model qualifies the robustness of a privacy-preserving algorithm under security attacks. An adversary usually works by colluding a number of nodes to conduct certain malicious behaviours, such as learning the private data and manipulating the outputs of the computations. These colluded nodes will be referred to as corrupted nodes and while the others will be referred to as honest nodes. Here we consider two general adversary models that are often encountered in real applications: passive and eavesdropping. In the former case, all nodes follow the instructions of the algorithm but they are curious about knowing the private data held by other honest nodes. The eavesdropping adversary, either internal or external, aims to infer the private data by eavesdropping the communication channels between honest nodes. This adversary has not received much attention in privacy-preserving distributed computation as it is commonly solved by assuming securely encrypted communication channels [14]. Encryption, however, incurs high computational complexity which is particularly cumbersome using iterative algorithms such as the ones we are using here, because communication channels are used many times. In this paper, we alleviate this problem and assume all the communication is done through non-secure channels except for the initialization.

D. Privacy-preserving distributed least squares

Combining things together, we conclude that there are two key requirements to be satisfied simultaneously:

1) Output correctness: all nodes are able to obtain the optimum solution $x^* = \arg\min_x \frac{1}{2}\|y - Qx\|_2^2$ when the algorithm converges.
2) Individual privacy: the concerned private data (Q_i, y_i) held by each node is protected from being revealed to others against both passive and eavesdropping adversaries, throughout the whole algorithm execution.

III. Primal-dual method of multipliers

We use PDMM as an example to explain the main idea of subspace perturbation. PDMM, like ADMM, is a distributed optimizer for solving constrained convex optimization problems. As an instance of Peaceman-Rachford splitting of the extended dual problem (see [13] for details), PDMM is characterised by a faster convergence rate compared to ADMM. The update equations of PDMM are given by

$$\begin{aligned} x^{(k+1)} &= \arg\min_x \big(f(x) + \lambda^{(k)^T} PCx + \frac{c}{2}\|Cx + PCx^{(k)}\|_2^2\big), \\ \lambda^{(k+1)} &= P\lambda^{(k)} + c(Cx^{(k+1)} + PCx^{(k)}), \end{aligned} \tag{2}$$

where $f(x)$ denotes the objective function to be minimised, k the iteration index, $x^{(k)} \in \mathbb{R}^n$ is the primal variable, $\lambda^{(k)} \in \mathbb{R}^{2m}$ the dual variable, $P \in \mathbb{R}^{2m\times 2m}$ a symmetric permutation matrix which exchanges the first m with the last m rows and $C \in \mathbb{R}^{2m\times n}$ a matrix related to the incidence matrix B. The constant $c > 0$ controls the convergence rate. The vector λ contains the dual variables for the constraints; there are two dual variables $\lambda_{i|j}$ and $\lambda_{j|i}$, one for each node i and j, for

each edge $(i,j) \in E$; where $\lambda(l) = \lambda_{i|j}$ and $C_{li} = B_{i|j}$ if and only if $e_l = (i,j) \in E$ and $i < j$, and $\lambda(l+m) = \lambda_{i|j}$, $C_{(l+m)i} = B_{i|j}$ if and only if $e_l = (i,j) \in E$ and $i > j$. Note that $C + PC = [B^T\, B^T]^T$ and $\forall (i,j) \in E : \lambda_{j|i} = \big(P\lambda\big)_{i|j}$.

The λ-updates of two successive iterations is given by

$$\lambda^{(k+2)} = \lambda^{(k)} + c(Cx^{(k+2)} + 2PCx^{(k+1)} + Cx^{(k)}), \quad (3)$$

as $P^2 = I$. Let $H = \mathrm{ran}(C) + \mathrm{ran}(PC)$ and $H^\perp = \mathrm{null}(C^T) \cap \mathrm{null}((PC)^T)$ where $\mathrm{ran}(\cdot)$ and $\mathrm{null}(\cdot)$ denote the range and nullspace, respectively. Note that $[C, PC] \in \mathbb{R}^{2m \times 2n}$ can be viewed as an incidence matrix of a new graph having $2n$ nodes and $2m$ edges. Therefore, we have $\dim(H) \leq 2n-1$ and thus $H^\perp$ is always non-empty. Let Π_H denote the orthogonal projection onto H. From (3) we can see that every two λ-updates only affect $\Pi_H\lambda \in H$ and leave $(I-\Pi_H)\lambda \in H^\perp$ unchanged. As a consequence, the component $(I-\Pi_H)\lambda$ will not converge and only be permuted every iteration. We can thus divide the dual variable $\lambda^{(k)}$ into two parts given by

$$\lambda^{(k)} = \Pi_H\lambda^{(k)} + \begin{cases} (I-\Pi_H)\lambda^{(0)}, & k \text{ even}, \\ P\,(I-\Pi_H)\,\lambda^{(0)}, & k \text{ odd}. \end{cases} \quad (4)$$

It is proven in [13] that $\Pi_H\lambda^{(k)}$ converges to the optimum λ^* given by

$$\lambda^* = -\begin{pmatrix} C^T \\ (PC)^T \end{pmatrix}^\dagger \begin{pmatrix} \nabla f(x^*) + cC^TCx^* \\ \nabla f(x^*) + cC^TPCx^* \end{pmatrix} + cCx^*, \quad (5)$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudo inverse. We thus denote $\Pi_H\lambda$ and $(I-\Pi_H)\lambda$ as the converging and non-converging component of the dual variable, respectively. Similarly, H and $H^\perp$ are referred to as the converging subspace and non-converging subspace of PDMM. It is worthy to mention that this non-converging component $(I-\Pi_H)\lambda$ would not affect the x-update in (2) since $\lambda^T(I-\Pi_H)PC = 0$.

IV. Proposed approach

Having introduced PDMM, we will now proceed to describe the proposed approach. For the problem at hand, the PDMM updating functions for node i become

$$x_i^{(k+1)} = (Q_i^TQ_i + cd_iI)^{-1}(Q_i^Ty_i + \sum_{j\in\mathcal{N}_i}(cx_j^{(k)} - B_{i|j}\lambda_{j|i}^{(k)}))$$

$$\forall j \in \mathcal{N}_i: \quad \lambda_{i|j}^{(k+1)} = \lambda_{j|i}^{(k)} + cB_{i|j}\big(x_i^{(k+1)} - x_j^{(k)}\big), \quad (6)$$

whereas the update of dual variable $\lambda_{i|j}^{(k+1)}$ only depends on $\lambda_{j|i}^{(k)}$, $x_j^{(k)}$ and $x_i^{(k+1)}$, of which $\lambda_{j|i}^{(k)}$ and $x_j^{(k)}$ are local information held by node j. Therefore, $x_i^{(k+1)}$ is the only information needs to be transmitted by node i to its neighbours. After broadcasting $x_i^{(k+1)}$, all neighbouring nodes can construct $\lambda_{i|j}^{(k+1)}$ themselves and the dual variables do not need to be transmitted at all, except for the first iteration where the initialized $\lambda_{j|i}^{(0)}$s need to be transmitted.

Since $x_i^{(k+1)}$ is the only revealed information, by inspecting the x-update in (6) we can see that $x_i^{(k+1)}$ is dependent of node i's private data Q_i, y_i and the data $x_j^{(k)}, \lambda_{j|i}^{(k)}$ from its neighbours. We therefore propose to initialize the dual variables in a way such that the non-converging component $(I-\Pi_H)\lambda$ sufficiently perturbs the private data Q_i, y_i. Thus the private data cannot be inferred and meanwhile the primal variable will still converge to x^*, as long as there is at least one honest neighbouring node. In what follows we will give a formal proof of this claim.

A. Output correctness

As proved in [13], the primal variable $x^{(k+1)}$ is guaranteed to converge to x^* geometrically given arbitrary initialization $x^{(0)}$ and $\lambda^{(0)}$, thereby guaranteeing the output correctness.

B. Individual privacy

Now we turn to analyse the individual privacy under both passive and eavesdropping adversaries. Under the passive adversary model, let $\mathcal{V}_c$ and $\mathcal{V}_h$ denote the set of corrupted and honest nodes, respectively. Without loss of generality, assume the passive adversary attempts to infer the private data of honest node $i \in \mathcal{V}_h$. As mentioned earlier, as the only information transmitted from node i after initialization is the primal variable $x_i^{(k+1)}$, the problem thus becomes to analyse how much information about Q_i and y_i would the passive adversary obtain by observing $x_i^{(k+1)}$. Using (4) we can express $x_i^{(k+1)}$ as

$$(Q_i^TQ_i + cd_iI)^{-1}\Bigg(\sum_{j\in\mathcal{N}_i\cap\mathcal{V}_h}\big(cx_j^{(k)} - B_{i|j}\big(P^k\Pi_H\lambda^{(k)}\big)_{j|i}\big) - \sum_{j\in\mathcal{N}_i\cap\mathcal{V}_h} B_{i|j}\big(P^k(I-\Pi_H)\lambda^{(0)}\big)_{j|i} + Q_i^Ty_i + c_p\Bigg), \quad (7)$$

where $c_p = \sum_{j\in\mathcal{N}_i\cap\mathcal{V}_c}\big(cx_j^{(k)} - B_{i|j}\lambda_{j|i}^{(k)}\big)$ can be considered constant as it is known by the passive adversary. As $k\to\infty$, x^* will be known and $\Pi_H\lambda^{(k)} \to \lambda^*$ given by (5). Thus we conclude that, as long as $\mathcal{N}_i \cap \mathcal{V}_h \neq \emptyset$, we can perturb the private data by introduce noise in $(I-\Pi_H)\lambda^{(0)}$. More specifically, let $s_i^q = (Q_i^TQ_i + cd_iI)^{-1}$, $s_i^y = Q_i^Ty_i$ and $\lambda^{(0)}$ denote realizations of the random variables $\bar{S}_i^q$, $\bar{S}_i^y$ and $\bar{\Lambda}^{(0)}$, respectively. Note that $\bar{\Lambda}^{(0)}$ is independent of both $\bar{S}_i^q$ and $\bar{S}_i^y$ as the initialization of dual variables is independent of the inputs. From (7), we can see that the information leakage regarding to Q_i and y_i can be represented by the mutual information [15] $I(\bar{S}_i^q, \bar{X}_i^{(k+1)})$ and $I(\bar{S}_i^y, \bar{X}_i^{(k+1)})$. To analyse both of them we need the following result.

Proposition 1. *Consider the continuous random variables $\{\bar{X}_1,\dots,\bar{X}_n\}$ having mean and variance $\mu_{\bar{X}_i}$ and $\sigma^2_{\bar{X}_i}$, respectively. Let $\{\bar{Y}_1,\dots,\bar{Y}_n\}$ be independent random variables independent of $\{\bar{X}_1,\dots,\bar{X}_n\}$. That is, $I(\bar{X}_i,\bar{Y}_j) = 0$ for all $(i,j) \in \mathcal{V}$. Let $\bar{Z}_i = \bar{X}_i + \bar{Y}_i$ and $\bar{W}_i = \bar{X}_i\bar{Y}_i$, and let $\bar{Z}_i' = \bar{Z}_i/\sigma_{\bar{Z}_i}$ and $\bar{W}_i' = \bar{W}_i/\sigma_{\bar{W}_i}$ be the normalised variables having unit variance. We then have*

$$\lim_{\sigma^2_{\bar{Y}_i}\to\infty} I(\bar{X}_1,\dots,\bar{X}_n;\bar{Z}_1,\dots,\bar{Z}_n) = 0,$$

$$\lim_{\sigma^2_{\bar{Y}_i}\to\infty} I(\bar{X}_1,\dots,\bar{X}_n;\bar{W}_1,\dots,\bar{W}_n) = 0.$$

Proof.

$$
\begin{aligned}
&I(\bar{X}_1,\dots,\bar{X}_n;\bar{Z}_1,\dots,\bar{Z}_n)\\
&\quad= h(\bar{Z}_1,\dots,\bar{Z}_n) - h(\bar{Z}_1,\dots,\bar{Z}_n|\bar{X}_1,\dots,X_n)\\
&\quad\overset{\text{(a)}}{=} h(\bar{Z}_1,\dots,\bar{Z}_n) - h(\bar{Y}_1,\dots,\bar{Y}_n)\\
&\quad\overset{\text{(b)}}{=} \sum_{i=1}^{n} h\big(\bar{Z}_i|\bar{Z}_1,\dots,\bar{Z}_{i-1}\big) - \sum_{i=1}^{n} h\big(\bar{Y}_i\big)\\
&\quad\overset{\text{(c)}}{\leq} \sum_{i=1}^{n} h\big(\bar{Z}_i\big) - \sum_{i=1}^{n} h(\bar{Y}_i)\\
&\quad\overset{\text{(d)}}{=} \sum_{i=1}^{n} I(\bar{X}_i;\bar{Z}_i)\\
&\quad\overset{\text{(e)}}{=} \sum_{i=1}^{n} I(\bar{X}_i/\sigma_{\bar{Z}_i};\bar{Z}_i'),
\end{aligned}
$$

where $h(\cdot)$ denotes the differential entropy of the random variable, assuming it exists. Step (a) follows from $h(\bar{Z}_i|\bar{X}_i) = h(\bar{Y}_i)$, (b) follows from the chain rule for differential entropy and the fact that the $\bar{Y}_i$'s are independent random variables, (c) follows from the fact that conditioning decreases entropy, (d) follows from $h\big(\bar{Z}_i\big) - h(\bar{Y}_i) = h(\bar{Z}_i) - h(\bar{Z}_i|\bar{X}_i) = I(\bar{X}_i;\bar{Z}_i)$ and (e) holds as mutual information is invariant under scaling. As a consequence

$$
\begin{aligned}
\lim_{\sigma^2_{\bar{Y}_i}\to\infty} \sum_{i=1}^{n} I(\bar{X}_i;\bar{Z}_i) &= \lim_{\sigma_{\bar{Z}_i}\to\infty} \sum_{i=1}^{n} I(\bar{X}_i/\sigma_{\bar{Z}_i};\bar{Z}_i')\\
&= \sum_{i=1}^{n} I(0;\bar{Z}_i') = 0.
\end{aligned}
$$

For the case $\bar{W}_i = \bar{X}_i\bar{Y}_i$, we have

$$
\begin{aligned}
h(\bar{W}_i|\bar{X}_i) &= \int p(\bar{x}_i) h(\bar{W}_i|\bar{X}_i = \bar{x}_i) d\bar{x}_i\\
&= \int p(\bar{x}_i) h(\bar{x}_i\bar{Y}_i|\bar{X}_i = \bar{x}_i) d\bar{x}_i\\
&\overset{\text{(a)}}{=} \int p(\bar{x}_i) h(\bar{Y}_i) d\bar{x}_i = h(\bar{Y}_i),
\end{aligned}
$$

where (a) holds since the probability measure of the event $\bar{X}_i = 0$ is zero. Hence, the proof of our second claim goes along the same lines as the one presented above, and we conclude that

$$
\begin{aligned}
&\lim_{\sigma^2_{\bar{Y}_i}\to\infty} I(\bar{X}_1,\dots,\bar{X}_n;\bar{W}_1,\dots,\bar{W}_n)\\
&\qquad\leq \lim_{\sigma_{\bar{W}_i}\to\infty} \sum_{i=1}^{n} I(\bar{X}_i/\sigma_{\bar{W}_i};\bar{W}_i') = 0,
\end{aligned}
$$

thereby proving our claims. □

By applying Proposition 1 to $I(\bar{S}_i^q, \bar{X}_i^{(k+1)})$ and $I(\bar{S}_i^y, \bar{X}_i^{(k+1)})$, we conclude that both mutual information can be made arbitrarily small by increasing the variance of the random variable $(I - \Pi_H)\Lambda^{(0)}$. We thus have both $I(\bar{S}_i^q, \bar{X}_i^{(k+1)}) = 0$ and $I(\bar{S}_i^y, \bar{X}_i^{(k+1)}) = 0$ if

$$
\exists j \in \mathcal{N}_i \cap \mathcal{V}_h : \operatorname{var}\big(((I - \Pi_H)\Lambda^{(0)})_{j|i}\big) \to \infty. \tag{8}
$$

Hence, the proposed approach is able to achieve asymptotically perfect security.

Algorithm 1 Privacy-preserving distributed least squares based on PDMM

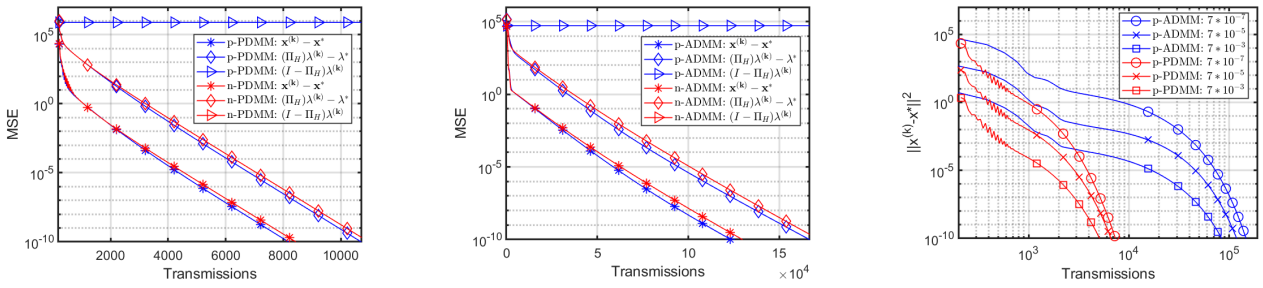
1: Every node $i \in \mathcal{V}$ initializes its primal variable arbitrarily, and initializes the dual variables with random numbers having sufficiently large variance (specified by the required privacy level).
2: Every node i sends the initialized dual variables $\lambda_{i|j}^{(0)}$ to its neighbours $j \in \mathcal{N}_i$ through securely encrypted channels.
3: **while** $\|x^{(k)} - x^*\|_2 <$ threshold **do**
4: Randomly activate a node, say node i, update its primal variable $x_i^{(k+1)}$ using the x-update in (6).
5: Node i broadcasts $x_i^{(k+1)}$ to its neighbours $j \in \mathcal{N}_i$ through non-secure channels.
6: Each neighbour uses $x_i^{(k+1)}$ to update the dual variable $\lambda_{i|j}^{(k+1)}$ based on the λ-update in (6).
7: **end while**

Now we consider an eavesdropping adversary. As we already proved that the transmitted primal variable does not contain information about the private data, the proposed method is also secure against eavesdropping. The communications can therefore be conducted in non-secure channels except for the first iteration where the initialized dual variables $\lambda^{(0)}$ should be communicated through secure channels. The details of the proposed approach are summarised in Algorithm 1.

Several remarks are in place here. Firstly, (8) requires $\lambda^{(0)} \cap H^\perp \neq \emptyset$. Recall that the non-converging subspace $H^\perp$ is non-empty, so that by randomly initializing the dual variables $\lambda^{(0)}$, we have $\lambda^{(0)} \cap H^\perp \neq \emptyset$ with probability 1. Secondly, it is important to note that the adversary does not have the knowledge of the subspace noise $(I - \Pi_H)\lambda^{(0)}$ as it does not know the converging subspace H, due to the fact that both the total number of nodes and the connectivity between the honest nodes are unknown to the adversary. Thirdly, although we proved that both $I(\bar{S}_i^q, \bar{X}_i^{(k+1)})$ and $I(\bar{S}_i^y, \bar{X}_i^{(k+1)})$ are zero if the inserted noise has infinitely large variance, in practical situation the noise variance will be finite. To quantify the amount of information leakage when dealing with finite variance noise, we consider the simple case of a random variable $\bar{Z} = \bar{X} + \bar{Y}$, where $\bar{X}$ and $\bar{Y}$ are independent Gaussian distributed random variables. For a Gaussian random variable with variance σ^2, the differential entropy is given by $\frac{1}{2}\log(2\pi e\sigma^2)$, so that $I(\bar{X};\bar{Z}) = h(\bar{Z}) - h(\bar{Y}) = \frac{1}{2}\log(1 + \sigma^2_{\bar{X}}/\sigma^2_{\bar{Y}})$. Hence, the information loss is only 0.007 bits if $\sigma^2_{\bar{Y}}/\sigma^2_{\bar{X}} = 100$ (the range of $\bar{Y}$ is approximately 10 times the range of $\bar{X}$). Lastly, we note that the proposed approach is also applicable to other distributed optimizers, e.g. ADMM, where the update equations of the dual variables have a similar structure as (2) and there also exists a non-converging subspace. To demonstrate this general applicability, in what follows we will show numerical results for both PDMM and ADMM.

V. Numerical results

We now evaluate the performance of the proposed algorithm by computer simulations. We simulated a random geometric



(a) (b) (c)

Fig. 1: Convergence of the primal variable, the converging component and non-converging component of the dual variable for two initializations of (a) PDMM and (b) ADMM. (c) Convergence of the primal variable of the proposed algorithm for ADMM and PDMM for three different privacy levels.

graph with $n = 20$ nodes, and set the wireless transmission radius as $\sqrt{2\frac{\log n}{n}}$ to obtain a connected graph with probability at least $1 - 1/n^2$ [16]. We set $N_i = 20, u = 10$ and generated all the entries of Q and y randomly according to a zero-mean, unit-variance Gaussian distribution.

Fig. 1a and 1b show the convergence behaviour of PDMM and ADMM, respectively (mean-squared error versus number of transmissions). The blue lines denote the proposed privacy-preserving approaches (p-PDMM and p-ADMM) where the dual variables are randomly initialized from a Gaussian distribution with variance 1000, while the red lines denote the non-private approaches (n-PDMM and n-ADMM) where the dual variables are initialized within the converging subspace, that is $\lambda^{(0)} \in H$. We can see that both $x^{(k)}$ and $\Pi_H \lambda^{(k)}$ converge to the optimum solution while $\big(I - \Pi_H\big)\lambda^{(k)}$ does not. Note that the lines with red triangle markers are not shown as $\big(I - \Pi_H\big)\lambda^{(k)} = 0$ in this case. Hence, the proposed approach is able to obfuscate the private data while not affecting the output correctness.

To inspect the performance of the proposed approach under different privacy levels, we considered three cases where the variances of the associated dual variables were set at 10, 100, and 1000, which corresponds to an approximated privacy loss of 7×10^{-3}, 7×10^{-5}, and 7×10^{-7} bits, respectively. As shown in Fig. 1c, for both PDMM and ADMM, the convergence rate is independent of the privacy level (note that the x-axis is on a log scale). This is because the convergence rate of these algorithms only depends on the graph topology and not on the initialization (the initial error does). Therefore, increasing the amount of noise will not affect the convergence rate but only results in a higher initial error.

VI. Conclusions

In this paper, we proposed a lightweight yet general convex optimization-based subspace perturbation method to achieve privacy-preserving distributed least squares. In particular, we show that the concerned private data can be protected by inserting noise in a particular subspace determined by the graph topology. The proposed approach is proven secure under both eavesdropping and passive adversaries. More specifically, the individual privacy of any honest node is protected as long as it has one honest neighbour and no securely encrypted channels are required except the initialization step. Additionally, it is able to achieve both privacy and accuracy simultaneously, and its convergence rate is independent of the privacy level.

References

[1] G. Giaconi, D. Gündüz, H. V. Poor, "Privacy-aware smart metering: Progress and challenges," *IEEE Signal Process. Mag., vol. 35, no. 6, pp. 59-78*, 2018.

[2] I. Damgård, V. Pastro, N. Smart, and S. Zakarias, "Multiparty computation from somewhat homomorphic encryption," in *Advances in Cryptology–CRYPTO, pp. 643–662*. Springer, 2012.

[3] I. Giacomelli, S. Jha, M. Joye, C. D. Page, and K. Yoon, "Privacy-preserving ridge regression with only linearly-homomorphic encryption," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Secur., pp. 243-261,*, 2018.

[4] A. Gascón, P. Schoppmann, B. Balle, M. Raykova, J. Doerner, S. Zahur, and D. Evans, "Privacy-preserving distributed linear regression on high-dimensional data," *in Proc. Priv. Enhancing Technol. no. 4, pp. 345–364*, 2017.

[5] V. Nikolaenko, U.Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft, "Privacy-preserving ridge regression on hundreds of millions of records," in *Proc. IEEE Symp. Security Privacy, pp. 334-348*, 2013.

[6] Y.R. Chen, A. Rezapour and W.-G Tzeng, "Privacy-preserving ridge regression on distributed data," *Inf. Sci., vol. 451, pp. 34-49,*, 2018.

[7] K. Tjell, I. Cascudo and R. Wisniewski, "Privacy preserving recursive least squares solutions," in *ECC, pp.3490–3495*, 2019.

[8] C. Dwork and J. Lei, "Differential privacy and robust statistics," in *Proc. 41st Annu. ACM Symp. Theory Comput., pp. 371-380*, 2009.

[9] D. Sarwate and K. Chaudhuri, "Signal processing and machine learning with differential privacy: Algorithms and challenges for continuous data," *IEEE Signal Process. Magazine, vol. 30, no. 5, pp. 86–94*, 2013.

[10] K. H. Degue and J. L. Ny, "On differentially private kalman filtering," in *Proc. IEEE Global Conf. Signal Inf. Process., pp. 487-491*, 2017.

[11] E. Nozari, P. Tallapragada, and J. Cortés, "Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design," *Automatica, vol. 81, pp. 221–231*, 2017.

[12] G. Zhang and R. Heusdens, "Distributed optimization using the primal-dual method of multipliers," *IEEE Trans. Signal Process., vol. 4, no. 1, pp. 173–187*, 2018.

[13] T. Sherson, R. Heusdens, W. B. Kleijn, "Derivation and analysis of the primal-dual method of multipliers based on monotone operator theory," *IEEE Trans. Signal Inf. Process. Netw., vol. 5, no. 2, pp 334-347*, 2018.

[14] D. Dolev, C. Dwork, O. Waarts, M. Yung, "Perfectly secure message transmission," *J. Assoc. Comput. Mach., vol. 40, no. 1, pp. 17-47,*, 1993.

[15] T. M. Cover and J. A. Tomas, *Elements of information theory*, John Wiley & Sons, 2012.

[16] J. Dall and M. Christensen, "Random geometric graphs," *Physical review E, vol. 66, no. 1, pp. 016121*, 2002.