FEDERATED OVER-AIR ROBUST SUBSPACE TRACKING FROM MISSING DATA

Praneeth Narayanamurthy, Namrata Vaswani and Aditya Ramamoorthy

Iowa State University

ABSTRACT

Robust Subspace Tracking with missing data (RST-miss) has been extensively studied in the past decade. In this work we study RST-miss to the setting where the data is federated and when the over-air data communication modality is used for information exchange between the K peer nodes and the central server. To the best of our knowledge, there is no existing work in the literature. To this end, we develop the first fast, and provable algorithm that solves RST-miss in a federated over-air setting. We corroborate our theoretical claims with extensive numerical simulations.

Index Terms— Robust Subspace Tracking, Robust PCA, Matrix Completion, Federated Learning

1. INTRODUCTION

Subspace tracking (ST) with missing data or outliers or both has been extensively studied in the last few decades [2–4]. ST with outlier data is commonly referred to as Robust ST (RST); it is the dynamic or "tracking" version of Robust PCA [5, 6]. Several algorithmic and theoretical approaches have been proposed to successfully solve these problems in the literature. In most of these works, the underlying assumption is that all the data is available at a single computational center. However, in many practical applications, the raw data is actually spread across numerous nodes geographically. In this paper, we consider the above problem in a federated setting, and when the over-air data communication modality [7] is used for information exchange between the K peer nodes and the central server.

(R)ST-miss has important applications in video analytics [8], social network activity learning [9] (anomaly detection) and recommendation system design [10] (learning time-varying low-dimensional user preferences from incomplete user ratings). The federated setting is most relevant for the latter two. At each time, each local node would have access to user ratings or messaging data from a subset of nearby users, but the subspace learning and matrix completion algorithm needs to use data from all the users.

Federated learning [11] refers to a distributed learning scenario in which individual nodes keep their data private but only share intermediate locally computed summary statistics with the central server at each algorithm iteration. The central server in turn, shares a global aggregate of these iterates with all the nodes. There has been a recent surge in works that solve machine learning problems in a federated setting, but all these assume a perfect channel between the peer nodes and the central server [12].

Advances in wireless communication technology now allow for (nearly) synchronous transmission by the various peer nodes and thus enable an alternate computation/communication paradigm for learning algorithms for which the aggregation step is a summation operation. In this alternate paradigm, the summation can be performed "over-air" and the summed aggregate or its processed version can be broadcasted to all the nodes [7, 13, 14]. Assuming K peer nodes, this paradigm is up to K-times more time- or bandwidth-efficient than the traditional mode. The drawback is that the transmitted data is corrupted by additive channel noise.

Related Work. While there are several papers that study PCA [10], Matrix Completion [22, 23], and ST-miss [24] in distributed settings, most of these come without provable guarantees, and do not account for iteration-noise. Other works that develop algorithms for the federated over-air aggregation setting include [7, 25] but they study different optimization problems. Finally, there is also some work on PCA in developing federated algorithms but without the over-air aggregation mode [26]. The only other existing works that study unsupervised learning algorithms with noisy algorithm iterations are [20, 21]; both these works study the noisy iteration version of the power method (PM) for computing the top r singular vectors of a given data matrix. In these works, noise is deliberately added to each algorithm iterate in order to ensure privacy of the data matrix.

Contributions. The main contribution of this work is a provable solution to the (R)ST-miss problem in the federated data setting when the data communication is done in the over-air mode. The main new challenge here is to develop approaches that are provably robust to additive noise in the algorithm iterates. This setting of noisy iterations has received little attention in literature as noted above. To the best of our knowledge, this is the first provable algorithm that studies (R)ST-miss in a federated, over-air paradigm. The main challenges here are (i) a design of an algorithm for this setting (this requires use of a federated over-air power method (FedOA-PM) for solving the PCA step); (ii) dealing with noise iterates due to the channel noise and (iii) dealing with mild asynchrony and channel fading. For (ii), the main work is in obtaining a modified result for PCA in sparse data-dependent noise solved via the FedOA-PM. For (iii), there exist a plethora of techniques within physical layer communications to circumvent these challenges [15]. The main idea is to use carefully designed pilot sequences to estimate the relative offset which can further be used to fix the asynchrony. For channel fading, the fading coefficients can also be estimated since the aforementioned pilot sequences are known. Thus, in this work, we only address iteration noise.

2. FEDERATED OVER-AIR PCA VIA THE POWER METHOD (PM)

We fist provide a result for subspace learning while obeying the federated data sharing constraints.

Problem setting. The goal of PCA (subspace learning) is to compute an r-dimensional subspace approximation in which a given data matrix $\boldsymbol{Z} \in \mathbb{R}^{n \times d}$ approximately lies. The k-th node observes a columns' sub-matrix $\boldsymbol{Z}_k \in \mathbb{R}^{n \times d_k}$. We have $\boldsymbol{Z} := [\boldsymbol{Z}_1, \cdots, \boldsymbol{Z}_k, \cdots, \boldsymbol{Z}_K] \in \mathbb{R}^{n \times d}$ with $d = \sum_{k=1}^{K} d_k$ and the goal of PCA is to find an $n \times r$ basis matrix $\boldsymbol{U}$ that minimizes

A full version of this paper is under review in IEEE Transactions on Signal Processing [1].

Algorithm 1 FedOA-PM: Federated Over-Air PM

Input: $\boldsymbol{Z}$ (data matrix), r (rank), L (# iterations), $\hat{\boldsymbol{U}}_0$ (optional initial subspace estimate)

1: K nodes, $\boldsymbol{Z}_k \in \mathbb{R}^{n\times d_k}$ local data at k-th node.
2: If no initial estimate provided, at central node, do $\tilde{\boldsymbol{U}}_0 \overset{i.i.d.}{\sim} \mathcal{N}(0,I)_{n\times r}$; $\hat{\boldsymbol{U}}_0 \leftarrow \tilde{\boldsymbol{U}}_0$, transmit to all K nodes.
3: **for** $l = 1,\dots,L$ **do**
4: At k-th node, for all $k \in [K]$, compute $\tilde{\boldsymbol{U}}_{k,l} = \boldsymbol{Z}_k\boldsymbol{Z}_k^\top \hat{\boldsymbol{U}}_{l-1}$
5: All K nodes transmit $\tilde{\boldsymbol{U}}_{k,l}$ synchronously to central node.
6: Central node receives $\tilde{\boldsymbol{U}}_l := \sum_k \tilde{\boldsymbol{U}}_{k,l} + \boldsymbol{W}_l$.
7: Central node computes $\hat{\boldsymbol{U}}_l\boldsymbol{R}_l \overset{QR}{\leftarrow} \boldsymbol{U}_l$
8: Central node broadcasts $\hat{\boldsymbol{U}}_l$ to all nodes
9: **end for**
10: At k-th node, compute $\tilde{\boldsymbol{U}}_{k,L+1} = \boldsymbol{Z}_k\boldsymbol{Z}_k^\top\hat{\boldsymbol{U}}_L$
11: All K nodes transmit $\tilde{\boldsymbol{U}}_{k,L+1}$ synchronously to the central node.
12: Central node receives $\tilde{\boldsymbol{U}}_{L+1} := \sum_k \tilde{\boldsymbol{U}}_{k,L+1} + \boldsymbol{W}_{L+1}$
13: Central node computes $\hat{\boldsymbol{\Lambda}} = \hat{\boldsymbol{U}}_L^\top\tilde{\boldsymbol{U}}_{L+1}$ and its top eigenvalue, $\hat{\sigma}_1 = \lambda_{\max}(\hat{\boldsymbol{\Lambda}})$.

Output: $\hat{\boldsymbol{U}}_L$, $\hat{\sigma}_1$.

$\|\boldsymbol{Z} - \boldsymbol{U}\boldsymbol{U}^\top\boldsymbol{Z}\|_F^2$. As is well known, the solution, $\boldsymbol{U}$, is given by the top r eigenvectors of $\boldsymbol{Z}\boldsymbol{Z}^\top$. Thus the goal is to estimate the span of $\boldsymbol{U}$ in a federated over-air (FedOA) fashion.

Federated Over-Air Power Method (FedOA-PM). The simplest algorithm for computing the top eigenvectors is the Power Method (PM) [27]. The distributed PM is well known, but most previous works assume the iteration-noise-free setting, e.g., see the review in [10]. On the other hand, there is recent work that studies the iteration-noise-corrupted PM [20, 21] but in the centralized setting. The vanilla PM estimates $\boldsymbol{U}$ by iteratively updating $\tilde{\boldsymbol{U}}_l = \boldsymbol{Z}\boldsymbol{Z}^\top\hat{\boldsymbol{U}}_{l-1}$ followed by QR decomposition to get $\hat{\boldsymbol{U}}_l$. FedOA-PM approximates this computation as follows. At iteration l, each node k computes $\tilde{\boldsymbol{U}}_{k,l} := \boldsymbol{Z}_k\boldsymbol{Z}_k^\top\hat{\boldsymbol{U}}_{l-1}$ and synchronously transmits it to the central server which receives the sum corrupted by channel noise, i.e., it receives

$$\tilde{\boldsymbol{U}}_l := \sum_{k=1}^K \tilde{\boldsymbol{U}}_{k,l} + \boldsymbol{W}_l = \boldsymbol{Z}\boldsymbol{Z}^\top\hat{\boldsymbol{U}}_{l-1} + \boldsymbol{W}_l.$$

since $\sum_k \boldsymbol{Z}_k\boldsymbol{Z}_k^\top = \boldsymbol{Z}\boldsymbol{Z}^\top$. Here $\boldsymbol{W}_l$ is the channel noise, and in this work we assume that each entry of $\boldsymbol{W}_l$ is i.i.d. zero-mean Gaussian with variance σ^2. The central server then computes a QR decomposition of $\tilde{\boldsymbol{U}}_l$ to get a basis matrix $\hat{\boldsymbol{U}}_l$ which is broadcast to all the K nodes for use in the next iteration. We summarize this complete FedOA-PM algorithm in Algorithm 1. If no initialization is available, it starts with a random initialization. When we use FedOA-PM for subspace tracking in the next section, the input will be the subspace estimate from the previous time instant.

We use σ_i to denote the i-th largest eigenvalue of $\boldsymbol{Z}\boldsymbol{Z}^\top$, i.e., $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n \geq 0$. We have the following guarantee.

Lemma 2.1 (FedOA-PM)**.** *Consider Algorithm 1. Pick the desired final accuracy $\epsilon \in (0, 1/3)$. Assume that, at each iteration, the channel noise $\boldsymbol{W}_l \overset{i.i.d.}{\sim} \mathcal{N}(0,\sigma_c^2)$ with (i) $\sigma_c < \epsilon\sigma_r/(5\sqrt{n})$ and (ii) $R := \sigma_{r+1}/\sigma_r < 0.99$.*

When using random initialization, if the number of iterations, $L = \Omega\left(\frac{1}{\log(1/R)}\log\left(\frac{nr}{\epsilon}\right)\right)$. then, with probability at least $0.9 - L\exp(-cr)$, $\mathrm{dist}(\boldsymbol{U},\hat{\boldsymbol{U}}_L) \leq \epsilon$.

When using an available initialization with $\mathrm{dist}(\hat{\boldsymbol{U}}_0,\boldsymbol{U}) < \epsilon_0$, if $L = \Omega\left(\frac{1}{\log(1/R)}\log\left(\frac{1}{\epsilon\sqrt{1-\epsilon_0^2}}\right)\right)$, then, with probability at least $1 - L\exp(-cr)$, $\mathrm{dist}(\boldsymbol{U},\hat{\boldsymbol{U}}_L) \leq \epsilon$.

Lemma 2.1 is similar to the one proved in [20, 21] for private PM but with a few key differences which are discussed in the long version [1]. We also provide a guarantee for the convergence of the maximum eigenvalue (Lines 10 − 13 of Algorithm 1) below.

Lemma 2.2 (FedOA-PM: Maximum eigenvalue)**.** *Let σ_i be the i-th largest eigenvalue of $\boldsymbol{Z}\boldsymbol{Z}^\top$. Under the assumptions of Lemma 2.1, $\hat{\sigma}_1$ computed in line 13 of Algorithm 1 satisfies*

$$(1-4\epsilon^2)\sigma_1 - \epsilon^2\sigma_{r+1} - \epsilon\sigma_r \leq \hat{\sigma}_1 \leq (1+\epsilon)\sigma_1$$

To our best knowledge, Lemma 2.2 has not been proved in earlier work. This result is useful because thresholding the top eigenvalue of an appropriately defined matrix is typically used for subspace change detection, see for example [18].

3. FEDERATED OVER-AIR RST-MISS

Problem setting. In this section, we use α_k to denote the number of data points at node k at time t and $\alpha := \sum_k \alpha_k$ to denote the total number at time t. We do this to differentiate from d (in Sec. 2) which is used to indicate the total number of data vectors. Thus, at time t, $d = t\alpha$ and $d_k = t\alpha_k$. At time t and node k, we observe a possibly incomplete and noisy data matrix $\boldsymbol{Y}_{k,t}$ of dimension $n \times \alpha_k$ with the missing entries being replaced by a zero. This means the following: let $\tilde{\boldsymbol{L}}_{k,t}$ denote the unknown, complete, approximately low-rank matrix at node k at time t. Then

$$\boldsymbol{Y}_{k,t} = \mathcal{P}_{\Omega_{k,t}}(\tilde{\boldsymbol{L}}_{k,t} + \boldsymbol{G}_{k,t}) = \mathcal{P}_{\Omega_{k,t}}(\tilde{\boldsymbol{L}}_{k,t}) + \boldsymbol{S}_{k,t}$$

where $\boldsymbol{G}_{k,t}$'s are sparse outliers and $\boldsymbol{S}_{k,t} := \mathcal{P}_{\Omega_{k,t}}(\boldsymbol{G}_{k,t})$, and $\mathcal{P}_{\Omega_{k,t}}$ sets entries outside the set $\Omega_{k,t}$ to zero. The full matrix available from all nodes at time t is denoted $\boldsymbol{Y}_t := [\boldsymbol{Y}_{1,t}, \boldsymbol{Y}_{2,t}, \dots, \boldsymbol{Y}_{K,t}]$. This is of size $n \times \alpha$. The true (approximately) rank-r matrix $\tilde{\boldsymbol{L}}_t$ is similarly defined. Define the index sets $\mathcal{I}_{1,t} := [1, 2, \dots, \alpha_1]$, $\mathcal{I}_{2,t} := [\alpha_1 + 1, \alpha_1 + 2, \dots, \alpha_1 + \alpha_2]$ and so on. Denote the i-th column of $\boldsymbol{Y}_t$ by $\boldsymbol{y}_i$, $i = 1, 2, \dots, \alpha$. And with slight abuse of notation, we define (the matrix binary masks) $\Omega_{1,t} := [(\mathcal{M}_{1,t})^c, (\mathcal{M}_{2,t})^c, \cdots, (\mathcal{M}_{\alpha_1,t})^c]$, $\Omega_{2,t} := [(\mathcal{M}_{\alpha_1+1,t})^c, (\mathcal{M}_{\alpha_1+2,t})^c, \cdots, (\mathcal{M}_{\alpha_1+\alpha_2,t})^c]$ and so on where $\mathcal{M}_{i,t}$ is the set of missing entries in column i of the data matrix at time t, $(\mathcal{M}_{i,t})^c$ is its complement w.r.t $[n]$. Thus, the observations satisfy

$$\boldsymbol{y}_i = \mathcal{P}_{\mathcal{M}_{i,t}^c}(\tilde{\boldsymbol{\ell}}_i) + \boldsymbol{s}_i, \quad i \in \mathcal{I}_{k,t}, \quad k \in [K] \tag{1}$$

where $\boldsymbol{s}_i$ are sparse vectors with support $\mathcal{M}_{\mathrm{sparse},i}$. Notice that it is impossible to recover $\boldsymbol{g}_i$ on the set $\mathcal{M}_{i,t}$ and so by definition, $\mathcal{M}_{\mathrm{sparse},i}, \mathcal{M}_{i,t}$ are disjoint. Let $\boldsymbol{P}_t$ denote the ($n \times r$ dimensional) matrix of top r left singular vectors of $\tilde{\boldsymbol{L}}_t$. In general, our assumptions imply that $\tilde{\boldsymbol{L}}_t$ is only approximately rank r. We define the matrix of the principal subspace coefficients at time t as $\boldsymbol{A}_t := \boldsymbol{P}_t^\top\tilde{\boldsymbol{L}}_t$, the rank-$r$ approximation, $\boldsymbol{L}_t := \boldsymbol{P}_t\boldsymbol{P}_t^\top\tilde{\boldsymbol{L}}_t$ and the "noise" orthogonal to the span($\boldsymbol{P}_t$) as $\boldsymbol{V}_t := \tilde{\boldsymbol{L}}_t - \boldsymbol{L}_t$. With these definitions, for all $i \in \mathcal{I}_{k,t}$ and $k \in [K]$, we can equivalently express the measurements as follows

$$\boldsymbol{y}_i = \mathcal{P}_{\mathcal{M}_{i,t}^c}(\tilde{\boldsymbol{\ell}}_i) + \boldsymbol{s}_i = \boldsymbol{\ell}_i + \boldsymbol{z}_i + \boldsymbol{s}_i + \boldsymbol{v}_i$$

The goal is to track the subspaces $\boldsymbol{P}_t$ quickly and reliably, and hence also reliably estimate the columns of the rank r matrix $\boldsymbol{L}_t$, under the FedOA constraints given earlier. Our problem can also be understood as a dynamic (changing subspace) version of robust MC [28].

Algorithm. The algorithm consists of two parts: (a) obtain *an* estimate of the columns $\tilde{\boldsymbol{L}}_t$ using the previous subspace estimate $\hat{\boldsymbol{P}}_{t-1}$; and (b) use this estimated matrix $\hat{\boldsymbol{L}}_t$ to update the subspace estimate, i.e., obtain $\hat{\boldsymbol{P}}_t$ by r-SVD. The algorithm can be initialized via r-SVD if we assume that $\boldsymbol{Y}_1$ (data available at $t=1$) contains no outliers and if not, one would need to use a batch RPCA approach such as AltProj [6] to obtain the initial subspace estimate $\hat{\boldsymbol{P}}_1$.

In the federated setting (a) is done locally at each node, while (b) requires a Fed-OA algorithm for SVD which is done using Algorithm 1. If one were to consider a federated but noise-free setting, there would be no need for new analysis.

For step (a) (obtaining an estimate of $\tilde{\boldsymbol{L}}_t$ column-wise), we use the projected Compressive Sensing (CS) idea [3]. This relies on the slow-subspace change assumption. Let $\hat{\boldsymbol{P}}_{t-1}$ denote the subspace basis estimate from the previous time and let $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}^\top$. Projecting $\boldsymbol{y}_i$ orthogonal to $\hat{\boldsymbol{P}}_{t-1}$ helps mostly nullify $\boldsymbol{\ell}_i$ but gives projected measurements of the missing entries, $\boldsymbol{I}_{\mathcal{M}_i}\boldsymbol{I}_{\mathcal{M}_i}^\top\boldsymbol{\ell}_i$ and the sparse outliers, $\boldsymbol{s}_i$ as follows

$$\boldsymbol{\Psi}\boldsymbol{y}_i = \underbrace{\boldsymbol{\Psi}(\boldsymbol{s}_i - \boldsymbol{I}_{\mathcal{M}_i}\boldsymbol{I}_{\mathcal{M}_i}^\top\boldsymbol{\ell}_i)}_{\text{projected sparse vector}} + \underbrace{\boldsymbol{\Psi}(\boldsymbol{\ell}_i + \boldsymbol{v}_i)}_{\text{error}}$$

If the previous subspace estimate is good enough, and the noise is small, the error term above will be small. Now recovering the vector $\boldsymbol{s}_i - \boldsymbol{I}_{\mathcal{M}_i}\boldsymbol{I}_{\mathcal{M}_i}^\top\boldsymbol{\ell}_i$ is from $\boldsymbol{\Psi}\boldsymbol{y}_i$ is a problem of noisy compressive sensing with partial support knowledge (since we know $\mathcal{M}_i$). We first recover the support of $\boldsymbol{s}_i$ using the approach of [29], and then perform a least-squares based debiasing to estimate the magnitude of the entries. Following this, *an* estimate of the true data, $\hat{\boldsymbol{\ell}}_i$ is computed by subtraction from the observed data $\boldsymbol{y}_i$. We show in [1] that $\hat{\boldsymbol{\ell}}_i$ satisfies

$$\hat{\boldsymbol{\ell}}_i = \boldsymbol{\ell}_i - \boldsymbol{I}_{\hat{\mathcal{M}}_i}\left(\boldsymbol{\Psi}_{\hat{\mathcal{M}}_i}^\top\boldsymbol{\Psi}_{\hat{\mathcal{M}}_i}\right)^{-1}\boldsymbol{I}_{\hat{\mathcal{M}}_i}^\top\boldsymbol{\Psi}(\boldsymbol{\ell}_i + \boldsymbol{v}_i) + \boldsymbol{v}_i \quad (2)$$

Now we have $\hat{\boldsymbol{L}}_t := [\hat{\boldsymbol{L}}_{1,t}, \hat{\boldsymbol{L}}_{2,t}, \cdots, \hat{\boldsymbol{L}}_{K,t}]$ with $\hat{\boldsymbol{L}}_{k,t}$ available only at node k. To goal is to compute an estimate ($\hat{\boldsymbol{P}}_t$) of its top r left singular vectors while obeying the federated data sharing constraints. We implement this through FedOA-PM (Algorithm 1) with $\boldsymbol{Z}_k \equiv \hat{\boldsymbol{L}}_{k,t}$ being the data matrix at node k. We invoke FedOA-PM with an initial estimate $\hat{\boldsymbol{P}}_{t-1}$. This simple change allows the probability of success of the overall algorithm to be close to 1 rather than 0.9 which is what the result of Lemma 2.1 predicts. This result is obtained by carefully combining the results for PCA-SDDN in a centralized setting and for FedOA-PM.

Assumptions and Main Result. It is well known from the LRMC literature [8] that for guaranteeing correct matrix recovery, we need to assume incoherence of the left and right singular vectors of the matrix. We need a similar assumption on $\boldsymbol{P}_t$'s.

Definition 3.1 (μ-Incoherence of $\boldsymbol{P}_t$s)**.** *Assume that* $\max_t \max_{m\in[r]} \|\boldsymbol{P}_t^{(m)}\|_2^2 \le \frac{\mu r}{n}$ *where* $\boldsymbol{P}_t^{(m)}$ *denotes the* m*-th row of* $\boldsymbol{P}_t$ *and* $\mu \ge 1$ *is a constant (incoherence parameter).*

Since we study a tracking algorithm, we replace the standard right singular vectors' incoherence assumption with the following simple statistical assumption on the subspace coefficients $\boldsymbol{a}_i$.

Algorithm 2 Fed-OA-RSTMiss-NoDet

Input: $\boldsymbol{Y}, \mathcal{M}$
1: Parameters: $L \leftarrow C\log(1/\text{no-lev}), \omega_{supp}, \xi, \alpha$
2: **Init:** $\tau \leftarrow 1, j \leftarrow 1, \hat{\boldsymbol{P}}_1$
3: **for** $t > 1$ **do**
4: $\quad\hat{\boldsymbol{L}}_t \leftarrow$ FED-MODCS$(\boldsymbol{y}_i, \mathcal{I}_{k,t}, \mathcal{M}_i, \hat{\boldsymbol{P}}_{t-1})$
5: $\quad\hat{\boldsymbol{P}}_t \leftarrow$ FEDOA-PM$(\hat{\boldsymbol{L}}_t, r, L, \hat{\boldsymbol{P}}_{t-1})$
6: $\quad\hat{\hat{\boldsymbol{L}}}_t \leftarrow$ FED-MODCS$(\boldsymbol{y}_i, \mathcal{I}_{k,t}, \mathcal{M}_i, \hat{\boldsymbol{P}}_t)$ ▷ optional
7: **end for**
Output: $\hat{\boldsymbol{P}}$

Algorithm 3 Federated Modified Compressed Sensing

procedure FED-MODCS$(\boldsymbol{y}_i, \mathcal{I}_{k,t}, \mathcal{M}_i, \hat{\boldsymbol{P}}_{t-1})$
$\quad$**for all** node k, $i \in \mathcal{I}_{k,t}$ **do**
$\quad\quad\boldsymbol{\Psi} \leftarrow \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}^\top$
$\quad\quad\tilde{\boldsymbol{y}}_i \leftarrow \boldsymbol{\Psi}\boldsymbol{y}_i$
$\quad\quad\hat{\boldsymbol{s}}_{i,cs} \leftarrow \arg\min_{\boldsymbol{s}} \|(\boldsymbol{s})_{(\mathcal{M}_i)^c}\|_1$ s.t. $\|\tilde{\boldsymbol{y}}_i - \boldsymbol{\Psi}\boldsymbol{s}\| \le \xi$.
$\quad\quad\hat{\mathcal{M}}_i \leftarrow \mathcal{M}_i \cup \{j : |(\hat{\boldsymbol{s}}_{i,cs})_j| > \omega_{supp}\}$
$\quad\quad\hat{\boldsymbol{\ell}}_i \leftarrow \boldsymbol{y}_i - \boldsymbol{I}_{\hat{\mathcal{M}}_i}(\boldsymbol{\Psi}_{\hat{\mathcal{M}}_i})^\dagger\tilde{\boldsymbol{y}}_i$.
$\quad$**end for**
$\quad$**Output:** $\hat{\boldsymbol{L}}_t$
end procedure

Definition 3.2 (Statistical μ-Incoherence of $\boldsymbol{a}_i$s)**.** *Recall that* $\boldsymbol{a}_i = \boldsymbol{P}_t^\top\tilde{\boldsymbol{\ell}}_i$ *for all* $i \in \mathcal{I}_{k,t}, k \in [K]$*. Assume that the* $\boldsymbol{a}_i$*'s are zero mean; mutually independent; have identical diagonal covariance matrix* $\boldsymbol{\Lambda}$*, i.e., that* $\mathbb{E}[\boldsymbol{a}_i\boldsymbol{a}_i^\top] = \boldsymbol{\Lambda}$ *with* $\boldsymbol{\Lambda}$ *diagonal; and are bounded, i.e.,* $\max_i \|\boldsymbol{a}_i\|^2 \le \mu r\lambda^+$*, where* $\lambda^+ := \lambda_{\max}(\boldsymbol{\Lambda})$ *and* $\mu \ge 1$ *is a small constant. Also, let* $\lambda^- := \lambda_{\min}(\boldsymbol{\Lambda})$ *and* $f := \lambda^+/\lambda^-$.

If a few complete rows (columns) of the entries are missing, it is impossible to recover the underlying matrix. This can be avoided by either assuming bounds on the number of missing entries in any row and in any column, or by assuming that each entry is observed uniformly at random independent of all others. In this work we assume the former which is a weaker assumption.

Definition 3.3 (Bounded Missing Entry Fractions)**.** *Consider the* $n \times \alpha$ *observed matrix* $\boldsymbol{Y}_t$ *at timir* t*. We use max-miss-frac-col (max-miss-frac-row) to denote the maximum of the fraction of missing entries in any column (row) of this matrix.*

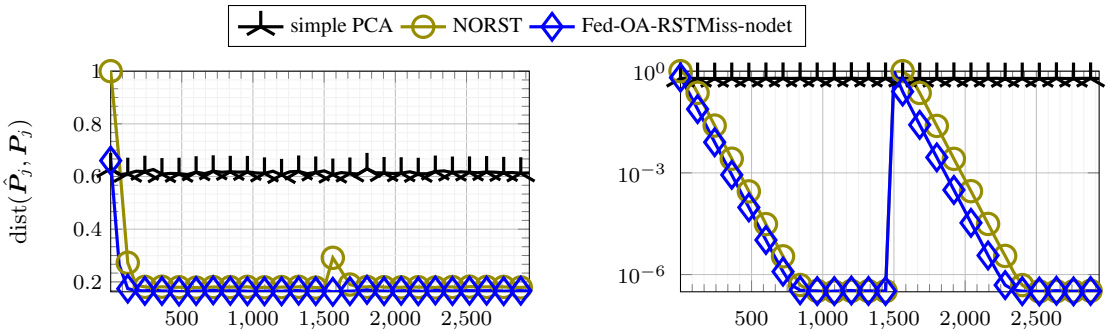
Definition 3.4 (Sparse outlier fractions)**.** *Consider the* $n \times \alpha$ *sparse outlier matrix* $\boldsymbol{S}_t := [\boldsymbol{S}_{1,t}, \ldots, \boldsymbol{S}_{K,t}]$ *at time* t*. We use max-out-frac-col (max-out-frac-row) to denote the maximum of the fraction of non-zero elements in any column (row) of this matrix. Also define* $s_{\min} = \min_{i\in\mathcal{I}_{k,t}} \min_{j\in\mathcal{M}_{\text{sparse},i}} |(\boldsymbol{s}_i)_j|$.

Finally, owing to the assumption that $\tilde{\boldsymbol{L}}_t$ is approximately low-rank, it follows that $\tilde{\boldsymbol{L}}_t - \boldsymbol{L}_t := \boldsymbol{V}_t$ is "small".

Definition 3.5 (Small, bounded, independent modeling error)**.** *Let* $\lambda_v^+ := \max_{i\in\mathcal{I}_{k,t}, k\in[K]} \|\mathbb{E}[\boldsymbol{v}_i\boldsymbol{v}_i^\top]\|$*. We assume that* $\lambda_v^+ < \lambda^-$, $\max_i \|\boldsymbol{v}_i\|^2 \le Cr\lambda_v^+$ *and* $\boldsymbol{v}_i$*'s are mutually independent over time.*

We have the following result.

Theorem 3.6 (Federated Robust Subspace Tracking NoDet)**.** *Consider Algorithm 2. Assume that* $\sqrt{\lambda_v^+/\lambda^-} :=$ *no-lev* ≤ 0.2*. Set* $L = C\log(1/\text{no-lev})$ *and* $\omega_{supp} = s_{\min}/2$, $\xi = s_{\min}/15$*. Assume that the following hold:*



(a) Small rotations at each time

(b) Piecewise Constant Subspace – Large

Fig. 1: Corroborating the claims of Theorem 3.6.

1. *At $t = 1$ we are given a $\hat{\boldsymbol{P}}_1$ s.t. $\mathrm{dist}(\boldsymbol{P}_1, \hat{\boldsymbol{P}}_1) \leq \epsilon_{\text{init}}$.*
2. **Incoherence:** *$\boldsymbol{P}_t$'s satisfy μ-incoherence, and $\boldsymbol{a}_i$'s satisfy statistical right μ-incoherence;*
3. **Missing Entries:** *max-miss-frac-col $\in O(1/\mu r)$, max-miss-frac-row $\in O(1)$;*
4. **Sparse Outliers:** *max-out-frac-col $\in O(1/\mu r)$, max-out-frac-row $\in O(1)$;*
5. **Channel Noise:** *the channel noise seen by each FedOA-PM iteration is mutually independent at all times, isotropic, and zero mean Gaussian with standard deviation $\sigma_c \leq$ no-lev$\lambda^- / 10\sqrt{n}$.*
6. **Subspace Model:** *The total data available at each time t, $\alpha \in \Omega(r \log n)$ and $\Delta_{tv} := \max_t \mathrm{dist}(\boldsymbol{P}_{t-1}, \boldsymbol{P}_t)$ s.t.*

$$0.3\epsilon_{\text{init}} + 0.5\Delta_{tv} \leq 0.28 \quad \textit{and}$$
$$C\sqrt{r\lambda^+}(0.3^{t-1}\epsilon_{\text{init}} + 0.5\Delta_{tv}) + \sqrt{r_v\lambda_v^+} \leq s_{\min}$$

then, with probability at least $1 - 10dn^{-10}$, for $t > 1$, we have

$$\begin{aligned}&\mathrm{dist}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)\\ &\leq \max(0.3^{t-1}\epsilon_{\text{init}} + \Delta_{tv}(0.3 + 0.3^2 ... + 0.3^{t-1}), \textit{no-lev})\\ &< \max(0.3^{t-1}\epsilon_{\text{init}} + 0.5\Delta_{tv}, \textit{no-lev})\end{aligned}$$

Also, at all times t, $\|\hat{\hat{\boldsymbol{\ell}}}_i - \boldsymbol{\ell}_i\| \leq 1.2 \cdot \mathrm{dist}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)\|\boldsymbol{\ell}_i\| + \|\boldsymbol{v}_i\|$ for all $i \in \mathcal{I}_{k,t}, k \in [K]$.

Discussion. Items 2-4 of Theorem 3.6 are necessary to ensure that the RST-miss and robust matrix completion problems are well posed [18, 28]. The initialization assumption of Theorem 3.6 is required due to the sensitivity of SVD to outliers. Without a "good initialization" Algorithm 2 cannot obtain good estimates of the sparse outliers since the noise in the sparse recovery step would be too large. One possibility to extend our result is to assume that there are no outliers at $t = 1$, i.e., $\boldsymbol{S}_1 = \boldsymbol{0}$ (see Remark 3.7). Item 5 is standard in the federated learning/differential privacy literature [20, 21] as without bounds on iteration noise, it is not possible to obtain a final estimate that is close to the ground truth. Finally, consider item 6: the first part is required to ensure that the projection matrices, $\boldsymbol{\Psi}$'s satisfy the restricted isometry property [29, 30] which is necessary for provable sparse recovery (with partial support knowledge). The second part of item 6 is an artifact of our analysis and arises due to the fact that it is hard to obtain element-wise error bounds for Compressive Sensing.

In Theorem 3.6 we assumed that we are given a good enough initialization. If however, $\boldsymbol{S}_1$ were 0, we have the following result.

Remark 3.7. *Under the conditions of Theorem 3.6, if $\boldsymbol{S}_1 = 0$, then all conclusions of Theorem 3.6 hold with the following changes*

1. *The number of iterations is set as $L = C\log(n/$no-lev$)$*
2. *The subspace model (item 6 satisfies all conditions with ϵ_{init} replaced by $0.01 \cdot 0.3$*
3. *The probability of success is now $0.9 - 10dn^{-10}$.*

4. NUMERICAL EXPERIMENTS

We corroborate the main result through numerical experiments. Due to space limitations, the complete details, and experiments are provided in the long version [1]. We consider two popular settings in the Subspace Tracking literature for the low-dimensional data: (a) small rotations at each time. We chose $n = 1000$, $d = 3000$, and $r = 30$. We generate data such that $\Delta_{tv} \approx 10^{-2}$; (b) piecewise-constant subspace model, with a large, abrupt change at $t_1 = 1500$. For sake of simplicity, we assume that $\boldsymbol{S} = 0$. We simulate the set of observed entries using a Bernoulli model where each element of the matrix is observed with probability 0.9. We compare with two baselines: (i) the first is a the naive approach of just computing the singular vectors as the PCA of the data; and (ii) NORST [18] (state-of-the-art theoretically) for centralized ST-miss. We implement FedOA-RST-Miss (Algorithm 2) with $\alpha = 60$. To simulate over-air communication, we replace the inbuilt `SVD` routine of MATLAB by a power method code snippet, and by adding iteration noise. In each iteration, we add i.i.d. Gaussian noise with variance 10^{-6}. The results are presented in Fig. 1. Notice that in both cases, Algorithm 2 works as well as NORST even though NORST cannot deal with iteration noise. All codes are availble at `https://github.com/praneethmurthy/distributed-pca`.

5. CONCLUSIONS

In this work we study the problem of Robust Subspace Tracking with missing entries in a federated, over-air setting. We developed the first provable algorithm dubbed FedOA-RST-miss to solve this problem while obeying the federated data sharing constraints. Our results show that under mild assumptions on underlying data, our algorithm recovers the underlying subspaces, and the sparse-outliers with high probability. We corroborate the theoretical claims with numerical experiments.

6. REFERENCES

[1] P. Narayanamurthy, N. Vaswani, and A. Ramamoorthy, "Federated over-the-air subspace learning and tracking from incomplete data," *arXiv preprint arXiv:2002.12873*, 2020.

[2] D. Zhang and L. Balzano, "Global convergence of a grassmannian gradient descent algorithm for subspace estimation," in *AISTATS*, 2016.

[3] C. Qiu, N. Vaswani, B. Lois, and L. Hogben, "Recursive robust pca or recursive sparse recovery in large but structured noise," *IEEE Trans. Info. Th.*, pp. 5007–5039, August 2014.

[4] C. Wang, Y. C. Eldar, and Y. M. Lu, "Subspace estimation from incomplete observations: A high-dimensional analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1240–1252, 2018.

[5] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, 2011.

[6] P. Netrapalli, U N Niranjan, S. Sanghavi, A. Anandkumar, and P. Jain, "Non-convex robust pca," in *Neural Information Processing Systems*, 2014.

[7] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," in *IEEE International Symposium on Information Theory*, 2019, pp. 1432–1436.

[8] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Found. of Comput. Math*, , no. 9, pp. 717–772, 2008.

[9] A. Zare, A. Ozdemir, M. A. Iwen, and S. Aviyente, "Extension of pca to higher order data structures: An introduction to tensors, tensor decompositions, and tensor pca," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1341–1358, 2018.

[10] S. X. Wu, H-T Wai, L. Li, and A. Scaglione, "A review of distributed algorithms for principal component analysis," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1321–1340, 2018.

[11] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[12] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[13] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *arXiv preprint arXiv:1907.09769*, 2019.

[14] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, 2020.

[15] D. Tse and P. Viswanath, *Fundamentals of wireless communication*, Cambridge university press, 2005.

[16] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Transactions on Signal Processing*, December 2013.

[17] P. Narayanamurthy and N. Vaswani, "Provable dynamic robust pca or robust subspace tracking," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1547–1577, 2019.

[18] P. Narayanamurthy, V. Daneshpajooh, and N. Vaswani, "Provable subspace tracking from missing data and matrix completion," *IEEE Transactions on Signal Processing*, pp. 4245–4260, 2019.

[19] A. Gonen, D. Rosenbaum, Y. C. Eldar, and S. Shalev-Shwartz, "Subspace learning with partial information," *Journal of Machine Learning Research*, vol. 17, no. 52, pp. 1–21, 2016.

[20] M. Hardt and E. Price, "The noisy power method: A meta algorithm with applications," in *Advances in Neural Information Processing Systems*, 2014, pp. 2861–2869.

[21] M-F. Balcan, S. S. Du, Y. Wang, and A. W. Yu, "An improved gap-dependency analysis of the noisy power method," in *Conference on Learning Theory*, 2016, pp. 284–309.

[22] C. Teflioudi, F. Makari, and R. Gemulla, "Distributed matrix completion," in *International Conference on Data Mining*. IEEE, 2012, pp. 655–664.

[23] L. Mackey, A. Talwalkar, and M. I. Jordan, "Distributed matrix completion and robust factorization," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 913–960, 2015.

[24] Y. Kopsinis, S. Chouvardas, and S. Theodoridis, "Distributed robust subspace tracking," in *European Signal Processing Conference (EUSIPCO)*, 2015, pp. 2531–2535.

[25] D. Gunduz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[26] A. Grammenos, R. Mendoza-Smith, C. Mascolo, and J. Crowcroft, "Federated pca with adaptive rank estimation," *arXiv preprint arXiv:1907.08059*, 2019.

[27] G. H. Golub and C. F. Van Loan, "Matrix computations," *The Johns Hopkins University Press, Baltimore, USA*, 1989.

[28] Y. Cherapanamjeri, K. Gupta, and P. Jain, "Nearly-optimal robust matrix completion," *ICML*, 2016.

[29] N. Vaswani and W. Lu, "Modified-cs: Modifying compressive sensing for problems with partially known support," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4595–4607, 2010.

[30] E. Candes, "The restricted isometry property and its implications for compressed sensing," *C. R. Math. Acad. Sci. Paris Serie I*, 2008.