Secure Algorithms for Vertically Federated Multi-Task Representation Learn

2025 IEEE International Symposium on Information Theory

Ankit Pratap Singh, Namrata Vaswani

Department of Electrical and Computer Engineering
Iowa State University

Problem Setting: Multi-task linear representation learning

Learn a low rank $r \ll n, q$ matrix $\mathbf{\Theta}^* \in \Re^{n \times q}$ from noisy measurements of the form

$$\mathbf{y}_k := \mathbf{X}_k \theta_k^* + \mathbf{v}_k, k \in [q].$$

- $\mathbf{y}_k \in \Re^m, \mathbf{X}_k \in \Re^{m \times n}$ is the training data for task k.
- $\mathbf{v}_k$ is the modeling error/noise.
- $\mathbf{X}_k$ are "random Gaussian" matrices which are independent and identically distributed (i.i.d.) over k.
- Noise $\mathbf{v}_k$ is independent of $\mathbf{X}_k$ and each entry of it is i.i.d. zero mean Gaussian with variance $\sigma_{\mathbf{v}}^2$.

Problem Setting

Because of the LR model, it is possible to recover $\mathbf{\Theta}^*$ even with $m < n$.

$$\mathbf{\Theta}^* = \mathbf{U}^* \mathbf{B}^*.$$

Where $\mathbf{U}^*$ is an $n \times r$ matrix with orthonormal columns, and $\mathbf{B}^*$ is an $r \times q$ matrix.

The learned representation in this case is an estimate of the column span of $\mathbf{U}^*$ (equivalently of $\mathbf{\Theta}^*$).

Solving this problem requires solving

$$\min_{\substack{\mathbf{U} \in \Re^{n \times r} \\ \mathbf{B} \in \Re^{r \times q}}} f(\mathbf{U}, \mathbf{B}) := \min_{\substack{\mathbf{U} \in \Re^{n \times r} \\ \mathbf{B} \in \Re^{r \times q}}} \sum_k \|\mathbf{y}_k - \mathbf{X}_k \mathbf{U} \mathbf{b}_k - \mathbf{v}_k\|_2^2$$

Vertical Federation

- Different nodes contain data for different subsets of tasks.
- Assume there are a total of L nodes.
- Let $\mathcal{S}_\ell, \ell \in [L]$ be a partition of $[q] := \{1, 2, \ldots, q\}$ such that $|\mathcal{S}_\ell| \geq q/L > r$ for all ℓ.
- Node ℓ has data $\mathbf{y}_k, \mathbf{X}_k$, for $k \in \mathcal{S}_\ell$.

Byzantine Attacks [1]

Byzantine attack is a "model update poisoning" attack where

1. It knows the full state of the center and every node (data and algorithm, including all algorithm parameters).
2. Different Byzantine adversaries can also collude.
3. They cannot modify the outputs of the other (non-Byzantine) nodes or of the center, or delay communication.

Byzantine nodes can thus design the worst possible attacks at each algorithm iteration.

[1] Mhamdi et al., The hidden vulnerability of distributed learning in byzantium, ICML, 2018

AltGDmin – intro

Assuming Right singular vectors' (RSV) incoherence[2] AltGDmin[3], a fast and communication-efficient GD-based algorithm was introduced for solving the problem in no-noise, and no-attack setting.

[2]Assume that $\max_{k\in[q]} \|\mathbf{b}_k^*\| \leq \mu\sqrt{r/q}\sigma_{\max}(\mathbf{\Theta}^*)$ for a constant $\mu \geq 1$.

[3]Nayer & Vaswani, Fast and sample-efficient federated low rank matrix recovery from column-wise linear and quadratic projections

AltGDmin – complete algorithm

$$f(\mathbf{U}, \mathbf{B}) := \sum_k \|\mathbf{y}_k - \mathbf{X}_k \mathbf{U} \mathbf{b}_k\|_2^2$$

- Initialization: Initialize $\mathbf{U}$ for the GD step as top r left singular vectors of $\mathbf{\Theta}_{init}$ matrix.

AltGDmin – complete algorithm

$$f(\mathbf{U}, \mathbf{B}) := \sum_k \|\mathbf{y}_k - \mathbf{X}_k \mathbf{U} \mathbf{b}_k\|_2^2$$

$$\mathbf{y}_k = \mathbf{X}_k \theta_k = \mathbf{X}_k \mathbf{U} \mathbf{b}_k, k \in [q]$$

Alt-GD-Min (GD step): at each iteration $t \geq 1$, alternate b/w

- min for $\mathbf{B}$: keeping $\mathbf{U}$ fixed, update $\mathbf{B}$ by solving $\min_{\mathbf{B}} f(\mathbf{U}, \mathbf{B})$. Clearly, this minimization decouples across columns, making it a cheap least squares problem of recovering q different r length vectors.

$$\mathbf{B} \leftarrow \arg\min_{\tilde{\mathbf{B}}} f(\mathbf{U}, \tilde{\mathbf{B}}) \ \Leftrightarrow \mathbf{b}_k = (\mathbf{X}_k \mathbf{U})^\dagger \mathbf{y}_k, \ k \in [q]$$

- projected GD for $\mathbf{U}$: keeping $\mathbf{B}$ fixed, update $\mathbf{U}$ by a GD step followed by orthonormalizing its columns.

$$\mathbf{U}^+ \leftarrow \mathrm{QR}(\mathbf{U} - \eta \nabla_U f(\mathbf{U}, \mathbf{B}))$$

$\mathbf{U} \leftarrow \mathbf{U}^+$

Initialization

Estimate principal subspace $span(\mathbf{U}^*)$ of an unknown matrix $\mathbf{\Theta}^*$ in a federated setting, while being resilient to **Byzantine Attacks**.

1. $\mathbf{U}^*_{n \times r}$ denotes the top r eigenvectors of $\mathbf{\Theta}^*$.
2. **Federated Setting:** Each node $\ell \in [L]$ can compute $(\mathbf{\Theta}_{init})_\ell$ using the columns $k \in \mathcal{S}_\ell$ that it observes. This allows the node to estimate $\mathbf{U}^*$ as $\mathbf{U}_\ell$, which is formed by the top r eigenvectors of $(\mathbf{\Theta}_{init})_\ell$.

Subspace-Median[4]

Subspace median a Byzantine-resilient subspace estimation algorithm which can be used for initialization part.

Theorem (Subspace-Median)

For a $\tau < 0.4$, suppose that, for at least $(1-\tau)L$ $\mathbf{U}_\ell$'s

$$\Pr\left(\mathbf{SD}_F(\mathbf{U}^*, \mathbf{U}_\ell) \leq \delta\right) \geq 1 - p$$

then, with probability at least $1 - \exp(-L\psi(0.4-\tau, p))$,

$$\mathbf{SD}_F(\mathbf{U}^*, \mathbf{U}_{out}) \leq 23\delta.$$

[4] Singh & Vaswani, Byzantine-resilient federated pca and low rank column-wise sensing, IEEE TIT, 2024

Dealing with attacks: Geometric Median

GM Theorem[5]**:** Let $\{\mathbf{z}_1, ..., \mathbf{z}_L\}$ with each $\mathbf{z}_\ell \subseteq \Re^n$ denote L nodes output, and let $\mathbf{z}_{gm}$ denote exact Geometric Median. For a $\tau < 0.4$, suppose that, at least $(1-\tau)L$ $\mathbf{z}_\ell$'s satisfy,

$$\|\mathbf{z}_\ell - \tilde{\mathbf{z}}\| \leq \epsilon$$

then,

$$\|\mathbf{z}_{gm} - \tilde{\mathbf{z}}\| \leq 6\epsilon$$

How it is handling Byzantine attacks? The rest τL, $\mathbf{z}_\ell$'s can be of arbitrary value.

[5] Stanislav Minsker, Geometric median and robust estimation in Banach spaces, Bernoulli, 2015

Including probability argument: For a $\tau < 0.4$, suppose that, at least $(1-\tau)L$ $\mathbf{z}_\ell$'s satisfy,

$$\Pr\{\|\mathbf{z}_\ell - \tilde{\mathbf{z}}\| \leq \epsilon\} \geq 1 - p$$

Then, w.p. at least $1 - \exp(-L\psi(0.4-\tau, p))$,

$$\|\mathbf{z}_{gm} - \tilde{\mathbf{z}}\| \leq 6\epsilon$$

Here

$$\psi(a,b) := (1-a)\log\frac{1-a}{1-b} + a\log\frac{a}{b}$$

The GM is defined for vectors whose distance can be measured using the vector l_2 norm. To use it for matrices we can use Frobenius norm.

$$\|\mathbf{M}\|_F = \|\mathrm{vec}(\mathbf{M})\|_2$$

General Statement

Fix an $\alpha \in (\tau, 1/2)$, $\tau = \frac{L_{byz}}{L}$, suppose that, at least $(1-\tau)L$ $\mathbf{z}_\ell$'s satisfy,

$$\Pr\{\|\mathbf{z}_\ell - \tilde{\mathbf{z}}\| \leq \epsilon\} \geq 1 - p$$

Then, w.p. at least $1 - \exp(-L\psi(\alpha - \tau, p))$,

$$\|\mathbf{z}_{gm} - \tilde{\mathbf{z}}\| \leq C_\alpha \epsilon$$

Here

$$\psi(a, b) := (1-a)\log\frac{1-a}{1-b} + a\log\frac{a}{b},$$

and

$$C_\alpha = (1-\alpha)\sqrt{\frac{1}{1-2\alpha}}.$$

- For $\alpha = 0$, $C_\alpha = 1$
- For $\alpha \to \frac{1}{2}$, $C_\alpha \to \infty$

Algorithm 1 Byz-Fed-AltGDmin-Learn: Complete algorithm

Nodes $\ell = 1, ..., L$
Compute $\mathbf{U}_{0\ell}$ which is the matrix of top r left singular vectors of $(\boldsymbol{\Theta}_{init})_\ell$.
Central Server (implements Subspace Median on $\mathbf{U}_{0\ell}$, $\ell \in [L]$)
Orthonormalize: $\mathbf{U}_{0\ell} \leftarrow QR(\mathbf{U}_{0\ell})$, $\ell \in [L]$
Compute $\mathcal{P}_{\mathbf{U}_{0\ell}} \leftarrow \mathbf{U}_{0\ell}\mathbf{U}_{0\ell}^\top$, $\ell \in [L]$
Compute GM: $\mathcal{P}_{gm} \leftarrow \mathrm{GM}\{\mathcal{P}_{\mathbf{U}_{0\ell}}, \ell \in [L]\}$
Find $\ell_{best} = \arg\min_\ell \|\mathcal{P}_{\mathbf{U}_{0\ell}} - \mathcal{P}_{gm}\|_F$
Output $\mathbf{U}_0 = \mathbf{U}_{out} = \mathbf{U}_{\ell_{best}}$
for $t = 1$ to T **do**
    Nodes $\ell = 1, ..., L$
    Set $\mathbf{U} \leftarrow \mathbf{U}_{t-1}$
    $\mathbf{b}_k \leftarrow (\mathbf{X}_k\mathbf{U})^\dagger \mathbf{y}_k, \ \forall \ k \in \mathcal{S}_\ell$
    $\nabla_\ell \leftarrow \sum_{k \in \mathcal{S}_\ell} \mathbf{X}_k^\top (\mathbf{X}_k \mathbf{U} \mathbf{b}_k - \mathbf{y}_k)\mathbf{b}_k{}^\top$
    Central Server
    $\nabla_{GM} \leftarrow \mathrm{GM}(\nabla_\ell, \ell = 1, 2, \ldots L)$.
    Compute $\mathbf{U}^+ \leftarrow QR(\mathbf{U}_{t-1} - \eta \nabla_{GM})$
    return Set $\mathbf{U}_t \leftarrow \mathbf{U}^+$. Push $\mathbf{U}_t$ to nodes.
end for

Byzantine-resilient Vertically federated MTRL [6]

Bounded heterogeneity: $\max_{\ell,\ell' \in [L]} \|\mathbf{B}^*_\ell - \mathbf{B}^*_{\ell'}\|_F^2 \leq G_B^2 \sigma^{*2}_{\max}$

Theorem

(Byz-Fed-AltGDmin-Learn: Complete guarantee) Assume RSV incoherence, Bounded heterogeneity Assumption holds, and $\frac{L_{byz}}{L} < 0.4$. *If*

$$m\left(\frac{q}{L}\right) \gtrsim nr \cdot \max\left(r, \log\left(\frac{1}{\epsilon}\right), \frac{NSR}{\epsilon^2} \log\left(\frac{1}{\epsilon}\right)\right)$$

then, w.h.p. after $T = C\tilde{\kappa}^2 \log\left(\frac{1}{\epsilon}\right)$ *iterations,*

$$\mathbf{SD}_F(\mathbf{U}^*, \mathbf{U}_T) \leq \max(\epsilon, 21 C\tilde{\kappa}^2 G_B)$$

Here NSR is the noise to signal ratio $\mathrm{NSR} := \frac{\widetilde{q}\sigma_v^2}{{\sigma^*_{\min}}^2}$

[6] Singh, & Vaswani, Secure Algorithms for Vertically Federated Multi-Task Representation Learni, ISIT, 2025

Challenges: Non Identical data

Since

$$\mathbb{E}[\nabla_\ell(\mathbf{U}_{t-1}, \mathbf{B}_\ell)] = m(\boldsymbol{\Theta}_\ell - \boldsymbol{\Theta}^*{}_\ell)\mathbf{B}_\ell{}^\top = m(\mathbf{U}\mathbf{B}_\ell - \mathbf{U}^*\mathbf{B}^*{}_\ell)\mathbf{B}_\ell{}^\top$$

Therefore,

$$\mathbb{E}[\nabla_\ell(\mathbf{U}_{t-1}, \mathbf{B}_\ell)] \neq \mathbb{E}[\nabla_{\ell'}(\mathbf{U}_{t-1}, \mathbf{B}_{\ell'})]$$

Bounded heterogeneity Assumption

$$\max_{\ell,\ell' \in [L]} \|\mathbf{B}^*_\ell - \mathbf{B}^*_{\ell'}\|_F^2 \leq G_B^2 \sigma_{\max}^{*2}$$

This assumption in turn implies that, for all $\ell, \ell' \in [L]$,

$$\|\mathbf{\Theta}^*{}_\ell - \mathbf{\Theta}^*{}_{\ell'}\|_F^2 = \|\mathbf{U}^*\mathbf{B}^*_\ell - \mathbf{U}^*\mathbf{B}^*_{\ell'}\|_F^2 \leq G_B^2 \sigma_{\max}^{*2}$$

All past work for heterogeneous setting assumes a bound on the difference between gradients from different good nodes, at each algorithm iteration [Assumption 2][7], [Assumption 1][8].

[7] Data & Diggavi, Byzantine-resilient high-dimensional federated learning, IEEE TIT, 2023

[8] Allouah et al., Fixing by mixing: A recipe for optimal byzantine ml under heterogeneity, AISTATS, 2023

Byzantine-resilient Vertically federated LRMC [9]

Incoherence of $\mathbf{U}^*$: MTRL/LRCS problem, does not require incoherence of $\mathbf{U}^*$. In LRMC, we need to ensure incoherence of $\mathbf{U}$ at every iteration. This is hard because $\mathbf{U}$ is updated using possibly non-incoherent gradients from GM or Krum. To handle this, we introduce a filtering step.

Paper to be presented in ICML 2025

[9] Singh, Abbasi, & Vaswani, Byzantine-Resilient Federated Alternating Gradient Descent and Minimization for Partly-Decoupled Low Rank Matrix Learning, ICML, 2025

Algorithm 2 Byz-AltGDmin-LRMC

AltGDmin Initialization:
Nodes $\ell = 1, ..., L$
Calculate and Push $\mathbf{U}_{0\ell}$ to center
Central Server
Define set $\mathcal{I}_0 = \{\}$
for $\ell = 1$ to L **do**
    if $\|\mathbf{u}^j_{0\ell}\| \leq 1.5\mu\sqrt{\frac{r}{n}}$ for all $j \in [n]$ **then**
        Add ℓ to set $\mathcal{I}_0$
end for
$\mathbf{U}_0 \longleftarrow \mathrm{Byz-SubspaceEstimation}\{\mathbf{U}_{0\ell}\}_{\ell \in \mathcal{I}_0}$
Push $\mathbf{U}_0$ to nodes.

Algorithm 3 Byz-AltGDmin-LRMC

AltGDmin Iterations:

for $t = 1$ to T **do**

  Nodes $\ell = 1, ..., L$

  Calculate and Push ∇_ℓ to center

  Central Server

  Define set $\mathcal{I}_t = \{\}$

  for $\ell = 1$ to L **do**

    Compute $\mathbf{U}_{temp} \leftarrow \mathbf{U}_{t-1} - \eta\nabla_\ell$

    if $\|\mathbf{u}^j_{temp}\| \leq (1 - \frac{0.4}{\tilde{\kappa}^2})\|\mathbf{u}^j_{t-1}\| + 1.4\mu\sqrt{\frac{r}{n}}$ for all $j \in [n]$ **then**

      Add ℓ to set $\mathcal{I}_t$

  end for

  $\nabla_{Kr/GM} = \mathrm{Krum/GM}\{\nabla_\ell\}_{\ell \in \mathcal{I}_t}$

  Compute $\mathbf{U}_t \leftarrow QR(\mathbf{U}_{t-1} - \eta\nabla_{Kr/GM})$

  Push $\mathbf{U}_t$ to nodes.

end for

Output $\mathbf{U}_T$.

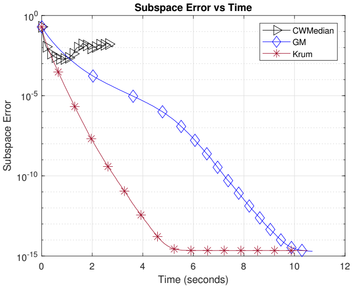
Notes on Robust Aggregators

- Compute cost for CWMed/CWTrim is smallest but its sample complexity is unreasonably high making it useless.
- Krum and GM have same sample complexity.
- GM compute cost using Accurate Median [10] is slightly less than Krum but it is an approximate algorithm i.e., we can compute GM with ϵ_{approx} error.
- However, Accurate Median is complex and to our best knowledge has no known experimental results.
- In practice, Weiszfeld's algorithm[11] is used to approximate GM. Weiszfeld's algorithm is known to converge, but the number of iterations is not specified.

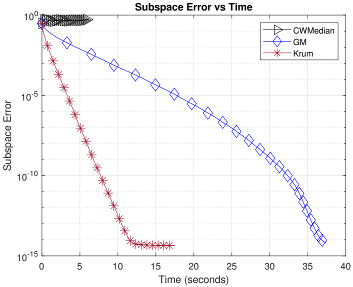
[10] Cohen et al., Geometric median in nearly linear time

[11] Endre Weiszfeld, Sur le point pour lequel la somme des distances den points donnés est minimum

Experiments



(a) **LRMC** with $n = 1000$, $q = 500$, $r = 3$, $L = 20$, and $L_{byz} = 8$.



(b) **LRCS** with $n = 1000$, $m = 50$, $q = 1000$, $r = 3$, $L = 20$, and $L_{byz} = 8$.

Figure 1: We compare Krum-AltGDmin, GM-AltGDmin, and CWMedian-AltGDmin for the different problems under the Reverse Gradient Attack.

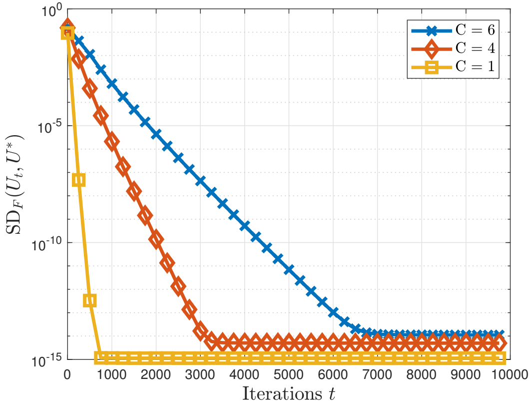


Figure 2: **Heterogeneity Effect: $\mathbf{SD}_F(\mathbf{U}_t, \mathbf{U}^*)$** vs Iteration t with $n = 200$, $q = 1000$, $r = 4$, $L = 10$, $L_{byz} = 2$, $p = 0.4$, Reverse Gradient Attack and using Krum

Thank You!

Questions?