Byzantine-resilient Alt-GD-Min for Federated Low Rank Matrix Learning

Namrata Vaswani

Professor of Electrical and Computer Engineering
Director, CyMath K-12 Math Support
Iowa State University, Ames, IA, USA
namrata@iastate.edu

Byzantine Resilient and Fast Federated Few-Shot Learning, ICML, 2024
Fast Sample-Efficient Federated Low Rank Matrix Recovery from Column-wise Linear and Quadratic Projections, IEEE Trans Info Theory, 2023
Fast Low Rank Compressive Sensing for Accelerated Dynamic MRI, IEEE Trans. Computational Imaging, 2023.

Collaborators

Current and former graduate students

- Dr. Seyedehsara Nayer
- Ankit Pratap Singh
- Dr. Ahmed Ali Abbasi
- Dr. Silpa Babu

Byzantine Resilient and Fast Federated Few-Shot Learning, ICML, 2024
Fast Sample-Efficient Federated Low Rank Matrix Recovery from Column-wise Linear and Quadratic Projections, IEEE Trans Info Theory, 2023
Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A simple proof, IEEE Trans Info Theory, 2024
Fast Low Rank Compressive Sensing for Accelerated Dynamic MRI, IEEE Trans. Computational Imaging, 2023.

Talk Outline

1. Introduction and General Alt-GD-Min Idea
2. AltGDmin for LR Column-wise compressive Sensing
3. Byzantine-Resilient AltGDmin
4. AltGDmin for other partly-decoupled LR problems
5. AltGDmin-MRI: AltGDmin-LRCS for Dynamic MRI
6. Early Math for Comm/SP/IT/STEM success

Introduction and General Alt-GD-Min Idea

Federated Learning: a type of Distributed Learning

- Multiple nodes/entities/clients collaborate in solving an ML problem
- Different subsets of data are available/acquired at distributed nodes
- Each node can only communicate with a central server or service provider ("center")

- Concerns:
  1. Communication-Efficiency: key concern for all distributed algo's
  2. Raw data privacy (sometimes)
  3. Resilience to attacks by adversaries, especially Byzantine attacks

Low-Rank (LR) Matrix Learning

Learn/recover an $n \times q$ matrix with rank $r \ll \min(n, q)$

$$\mathbf{X}^* = [\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_k^*, \ldots, \mathbf{x}_q^*]$$

from undersampled, incomplete, outlier-corrupted or entry-wise nonlinear.

- MRI image sequences – LR column-wise sensing
- video backgrounds – robust PCA
- product ratings – robust LR matrix completion
- correlated regression coefficients in multi-task learning – LR column-wise sensing
- changes to weights in LLM fine tuning (Low Rank Adaptation)

Federated setting: subsets of columns of $\mathbf{X}^*$ sensed at diff nodes

When/why Alternating GD and minimization (AltGDmin)?

Consider a (non-convex) optimization problem

$$\min_{\mathbf{Z}} f(\mathbf{Z})$$

Common opt solutions: after appropriate initialization, use

- Gradient Descent (GD): for any differentiable $f(\mathbf{Z})$

$$\hat{\mathbf{Z}}^+ \leftarrow \hat{\mathbf{Z}} - \eta \nabla_{\mathbf{Z}} f(\hat{\mathbf{Z}})$$

  converges slowly or may not; but low per-iteration cost typically

- AltMin: if $\mathbf{Z} = \{\mathbf{Z}_a, \mathbf{Z}_b\}$ s.t. min over $\mathbf{Z}_a$, keeping $\mathbf{Z}_b$ fixed, & vice versa, closed-form

$$\hat{\mathbf{Z}}_a^+ \leftarrow \arg\min_{\mathbf{Z}_b} f(\hat{\mathbf{Z}}_a, \mathbf{Z}_b), \quad \hat{\mathbf{Z}}_a^+ \leftarrow \arg\min_{\mathbf{Z}_a} f(\mathbf{Z}_a, \hat{\mathbf{Z}}_b^+)$$

  converges fast but high per-iteration cost typically

When/why Alternating GD and minimization (AltGDmin)?

Consider a (non-convex) optimization problem

$$\min_{\mathbf{Z}} f(\mathbf{Z})$$

Common opt solutions: after appropriate initialization, use

- Gradient Descent (GD): for any differentiable $f(\mathbf{Z})$

$$\hat{\mathbf{Z}}^+ \leftarrow \hat{\mathbf{Z}} - \eta \nabla_{\mathbf{Z}} f(\hat{\mathbf{Z}})$$

  converges slowly or may not; but low per-iteration cost typically

- AltMin: if $\mathbf{Z} = \{\mathbf{Z}_a, \mathbf{Z}_b\}$ s.t. min over $\mathbf{Z}_a$, keeping $\mathbf{Z}_b$ fixed, & vice versa, closed-form

$$\hat{\mathbf{Z}}_a^+ \leftarrow \arg\min_{\mathbf{Z}_b} f(\hat{\mathbf{Z}}_a, \mathbf{Z}_b), \quad \hat{\mathbf{Z}}_a^+ \leftarrow \arg\min_{\mathbf{Z}_a} f(\mathbf{Z}_a, \hat{\mathbf{Z}}_b^+)$$

  converges fast but high per-iteration cost typically

Alt-GD-Min: use if one of the two min's in AltMin is quick, say due to decoupling [Nayer & Vaswani, IEEE Trans. Info. Theory, 2023]

- often as fast per-iteration as GD and converges almost as fast as AltMin
- Improved overall speed and or communication-efficiency

AltGDmin for LR Column-wise compressive Sensing

Recover an $n \times q$ rank-r matrix $\mathbf{X}^* := [\mathbf{x}_1^*, \mathbf{x}_2^*, \dots \mathbf{x}_q^*]$ from

$$\mathbf{y}_k := \mathbf{A}_k \mathbf{x}_k^*, \ \ k = 1, 2, \dots, q$$

where $\mathbf{A}_k$ are known $m \times n$ matrices with $m \ll n$. Any LR $\mathbf{X}^*$ can be expressed as

$$\mathbf{X}^* = \mathbf{U}^* \mathbf{B}^*$$

with $\mathbf{U}^* \in \Re^{n \times r}$ being its matrix of top r singular vectors and $\mathbf{B}^* = \mathbf{U}^{*\top} \mathbf{X}^*$

Federation: Different columns $\mathbf{y}_k$ are observed at different nodes

Recover an $n \times q$ rank-r matrix $\mathbf{X}^* := [\mathbf{x}_1^*, \mathbf{x}_2^*, \dots \mathbf{x}_q^*]$ from

$$\mathbf{y}_k := \mathbf{A}_k \mathbf{x}_k^*, \ \ k = 1, 2, \dots, q$$

where $\mathbf{A}_k$ are known $m \times n$ matrices with $m \ll n$. Any LR $\mathbf{X}^*$ can be expressed as

$$\mathbf{X}^* = \mathbf{U}^* \mathbf{B}^*$$

with $\mathbf{U}^* \in \Re^{n \times r}$ being its matrix of top r singular vectors and $\mathbf{B}^* = \mathbf{U}^{*\top} \mathbf{X}^*$

Federation: Different columns $\mathbf{y}_k$ are observed at different nodes

Applications:

- Accelerated (under-sampled) dynamic MRI:
  $\mathbf{x}_k^*$: vectorized k-th image, $\mathbf{A}_k$: partial Fourier [Lingala et al],[Babu,Lingala,Vaswani, TCI,2023]
- Federated sketching of a set of similar images acquired at distributed nodes:
- Multi-task representation learning for few-shot learning a.k.a. Meta Learning

Factoring the unknown $\mathbf{X}$ as $\mathbf{X} = \mathbf{UB}$, we need to solve

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

(notice the decoupling over columns of $\mathbf{B}$)

- Initialize $\mathbf{U}$ – next slide
- Alt-GD-Min: alternate b/w
  - Min over $\mathbf{B}$

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

  - GD over $\mathbf{U}$ followed by orthonormalization

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

Nayer and Vaswani, Fast and Sample-Efficient Federated Low Rank Matrix Recovery from column-wise Linear and Quadratic Projections, IEEE Trans. Information Theory, 2023

Consider

$$\mathbf{X}_{0,full} = \frac{1}{m}\left[(\mathbf{A}_1^\top \mathbf{y}_1), ..., (\mathbf{A}_k^\top \mathbf{y}_k), ..., (\mathbf{A}_q^\top \mathbf{y}_q)\right]$$

For random Gaussian $\mathbf{A}_k$'s, $\mathbb{E}[\mathbf{A}_k^\top \mathbf{y}_k] = \mathbb{E}[\mathbf{A}_K^\top \mathbf{A}_k \mathbf{x}_k^*] = m\mathbf{x}_k^*$ and so

$$\mathbb{E}[\mathbf{X}_{0,full}] = m\mathbf{X}^*$$

Thus, if mq large enough, $\mathbf{X}_{0,full} \approx \mathbf{X}^*$ and so the same is true for their span of top r singular vectors.

Consider

$$\mathbf{X}_{0,full} = \frac{1}{m}\left[(\mathbf{A}_1^\top \mathbf{y}_1), ..., (\mathbf{A}_k^\top \mathbf{y}_k), ..., (\mathbf{A}_q^\top \mathbf{y}_q)\right]$$

For random Gaussian $\mathbf{A}_k$'s, $\mathbb{E}[\mathbf{A}_k^\top \mathbf{y}_k] = \mathbb{E}[\mathbf{A}_K^\top \mathbf{A}_k \mathbf{x}_k^*] = m\mathbf{x}_k^*$ and so

$$\mathbb{E}[\mathbf{X}_{0,full}] = m\mathbf{X}^*$$

Thus, if mq large enough, $\mathbf{X}_{0,full} \approx \mathbf{X}^*$ and so the same is true for their span of top r singular vectors.

But notice that $\mathbf{X}_{0,full} = \sum_{k=1}^{q}\sum_{i=1}^{m} \mathbf{a}_{ki}\mathbf{y}_{ki}\mathbf{e}_k{}^\top$

- summands heavy-tailed: a few large $\mathbf{y}_{ki}$'s can bias the average a lot
  - sub-exponential w/ worst-case sub-expo norm too large
- Fix: truncate – sum over only the "not-too-large" $\mathbf{y}_{ki}$'s
  - converts summands to sub-Gaussian r.v.s
    - ([Candes, Chen, NIPS'15] used truncation to convert heavier-tailed than sub-expo summands to sub-expo)

Nayer and Vaswani, Fast and Sample-Efficient Federated Low Rank Matrix Recovery from column-wise Linear and Quadratic Projections, IEEE Trans. Information Theory, 2023

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

- Initialize: Compute $\mathbf{U}$ as top r left singular vectors of

$$\mathbf{X}_{init} := \frac{1}{m}\left[(\mathbf{A}_1^\top \mathbf{y}_{1,trnc}), ..., (\mathbf{A}_k^\top \mathbf{y}_{k,trnc}), ..., (\mathbf{A}_q^\top \mathbf{y}_{q,trnc})\right]$$

  where $\mathbf{y}_{k,trnc} = \mathbf{y}_k \circ \mathbb{1}\{|\mathbf{y}_k| \le \sqrt{\alpha}\}$ and $\alpha = \tilde{C}\frac{1}{mq}\sum_{ki}|\mathbf{y}_{ki}|^2$

- Alt-GD-Min: at each iteration $t \ge 1$, alternate b/w
  - min for $\mathbf{B}$

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

  - GD for $\mathbf{U}$

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

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

Let $\mathcal{S}_\ell, \ell \in [L]$ be a partition of $[q]$. Node ℓ observes $\{\mathbf{y}_k, \mathbf{A}_k\}, k \in \mathcal{S}_\ell$

Let

$$f_\ell(\mathbf{U}, \mathbf{B}_\ell) := \sum_{k \in \mathcal{S}_\ell} \|\mathbf{y}_k - \mathbf{A}_k \mathbf{U} \mathbf{b}_k\|_2^2 \text{ where } \mathbf{B}_\ell := [\mathbf{b}_k, \ k \in \mathcal{S}_\ell]$$

Initialize: use federated power method to compute $\mathbf{U}$ by r-SVD on $\mathbf{X}_0$.

Alt-GD-Min: at each iteration $t \geq 1$,

- Each node ℓ:
  - compute $\mathbf{b}_k = (\mathbf{A}_k \mathbf{U})^\dagger \mathbf{y}_k, \ k \in \mathcal{S}_\ell$
  - compute $\nabla_\ell = \nabla_U f_\ell(\mathbf{U}, \mathbf{B}_\ell)$
  - transmit ∇_ℓ to center
- Center: GD for $\mathbf{U}$
  - GD and orthonormalize: $\mathbf{U}^+ \leftarrow \mathrm{QR}\left(\mathbf{U} - \eta \sum_\ell \nabla_\ell\right)$
  - $\mathbf{U} \leftarrow \mathbf{U}^+$ and broadcast to nodes

  Per-iteration communic: nr, time: $mqnr$

Guarantee[1]

Recover $\mathbf{X}^*$ ($n \times q$ with rank r) from $\mathbf{y}_k := \mathbf{A}_k \mathbf{x}_k^*$, $k \in [q]$, $\mathbf{A}_k$: i.i.d. r. Gaussian.

Theorem

Assume μ-incoherence of right singular vectors. Set $\eta = 0.4/{\sigma^*_{\max}}^2$, $T := \kappa^2 \log(1/\epsilon)$. If

$$mq \geq C_{\kappa,\mu} nr \max\left(r, \log(1/\epsilon)\right),$$

and $m \geq C \max(r, \log q, \log n) \log(1/\epsilon)$, then, w.p. at least $1 - Tn^{-10}$,

$$\mathrm{SD}(\mathbf{U}, \mathbf{U}^*) \leq \epsilon \;\; \textit{and} \;\; \|\mathbf{x}_k - \mathbf{x}_k^*\|_2 \leq \epsilon \|\mathbf{x}_k^*\|_2, \; \forall \; k \in [q]$$

Time: $mqnr \log(1/\epsilon)$. Communic: $nr \log(1/\epsilon)$. Samples: $\max(nr^2, nr \log(1/\epsilon))$

Notes

- Subspace Distance (SD): for "basis" matrices $\mathbf{U}_1, \mathbf{U}_2$, $\mathrm{SD}(\mathbf{U}_1, \mathbf{U}_2) := \|(\mathbf{I} - \mathbf{U}_1 {\mathbf{U}_1}^\top)\mathbf{U}_2\|_2$
- μ-incoherence of right sing vecs: $\max_k \|\mathbf{x}_k^*\|_2 \leq \mu \sqrt{r/q} \sigma^*_{\max}$

[1] Nayer and Vaswani, Fast and Sample-Efficient Federated Low Rank Matrix Recovery from column-wise Linear and Quadratic Projections, IEEE Trans. Info. Th., 2023]
Vaswani, Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A Simple Proof, IEEE Trans. Info Th., 2024

Why a new GD based approach is needed for LRCS [Nayer & Vaswani, IEEE Trans. Info. Theory, 2023][2]

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

- AltMin is expensive
  - min w.r.t. $\mathbf{U}$ is not decoupled.
- Problem decoupled column-wise but the two existing GD algo's are not
  - Need tight column-wise bound $\max_k \|\mathbf{x}_k - \mathbf{x}_k^*\|_2 \lesssim \delta_t \sqrt{r/q}\sigma_{\max}^*$ to bound gradient. But due to coupling in the algo's, one can only bound $\|\mathbf{X} - \mathbf{X}^*\|_F$
    - ★ Projected GD on $\mathbf{X}$, $\mathbf{X}^+ \leftarrow \mathcal{P}_r(\mathbf{X} - \eta \nabla_{\mathbf{X}} f(\mathbf{X}))$ [Cherapanamjeri et al,ICML'16] for LRMC
    - ★ Factorized GD: GD for $\tilde{f}(\mathbf{UB}) + \underbrace{\|\mathbf{U}^\top\mathbf{U} - \mathbf{B}\mathbf{B}^\top\|_F}_{\text{norm-balancing-term}}$, $\mathbf{X} = \mathbf{UB}$ [studied in [Xi et al,NeurIPS'16] for LRMC]
  - Cannot show decay of bound on gradient norm, w/ constant step size, under the desired sample complexity

[2] Nayer and Vaswani, Fast and Sample-Efficient Federated Low Rank Matrix Recovery from column-wise Linear and Quadratic Projections, IEEE Trans. Info. Theory, 2023

Existing results versus ours

	Sample Comp. $mq \gtrsim$	Time Comp.	Comm. Comp. per iter per node
Convex [Srinivasa et al'19]	$nr\frac{1}{\epsilon^4}$	$\mathrm{LinTime} \cdot r \cdot \frac{1}{\sqrt{\epsilon}}$	not clear
AltMin [Nayer, Vaswani, et al'19,'20,'22]	$nr^2 \log(\frac{1}{\epsilon})$	$\mathrm{LinTime} \cdot r \cdot \log^2(\frac{1}{\epsilon})$	$nr \log(\frac{1}{\epsilon})$
GD (fact-GD) [Xi et al,NeurIPS'16] for LRMC	unknown	$\mathrm{LinTime} \cdot r^2 \cdot \log(\frac{1}{\epsilon})$	nr
Projected GD [Cherapanamjeri et al,ICML'16] for LRMC	unknown	$\mathrm{LinTime} \cdot r \cdot \log^2(\frac{1}{\epsilon})$	nq
AltGDmin (proposed)	$nr \max(r, \log(\frac{1}{\epsilon}))$	$\mathrm{LinTime} \cdot r \cdot \log(\frac{1}{\epsilon})$	nr

Proof overall idea: bound initialization error using Wedin & sub-Gaussian Hoeffding

At iteration t, assume $\mathrm{SD}(\mathbf{U}^*, \mathbf{U}) \le \delta_t < \frac{0.1}{\sqrt{r}\kappa^2}$. Show $\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) \le (1 - c_0/\kappa^2)\delta_t$

- Update of $\mathbf{B}$: $\mathbf{b}_k = (\mathbf{A}_k\mathbf{U})^\dagger \mathbf{y}_k$ for all $k \in [q]$
  - Can show: w.h.p.,
$$\|\mathbf{b}_k - \mathbf{U}^\top \mathbf{U}^* \mathbf{b}_k^*\| \le 0.4\|(\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{U}^*\mathbf{b}_k^*\|$$
  - So,
    - ★ $\|\mathbf{x}_k - \mathbf{x}_k^*\| \le 1.4\|(\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{U}^*\mathbf{b}_k^*\| \le 1.4\delta_t\|\mathbf{x}_k^*\|$
    - ★ use this to show incoherence of $\mathbf{b}_k$s
    - ★ use this to lower and upper bound $\sigma_{\min}(\mathbf{X}), \sigma_{\max}(\mathbf{X})$.
- Update of $\mathbf{U}$: $\tilde{\mathbf{U}}^+ = \mathbf{U} - \eta\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})$, $\mathbf{U}^+ = \mathrm{QR}(\tilde{\mathbf{U}}^+)$
  - Since $\mathbf{U}^+ = \tilde{\mathbf{U}}^+(\mathbf{R}^+)^{-1}$ and $\sigma_{\min}(\mathbf{R}^+) = \sigma_{\min}(\tilde{\mathbf{U}}^+)$,
$$\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) \le \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})\tilde{\mathbf{U}}^+\|}{\sigma_{\min}(\tilde{\mathbf{U}}^+)}$$
$$\le \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})(\mathbf{U} - \eta\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})]\| - \|\mathrm{diff}\|)}{1 - \eta\|\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})]\| - \eta\|\mathrm{diff}\|}$$
  where $\mathrm{diff} := \nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B}) - \mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})]$.
- Simplify this and set η to show that $\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) \le (1 - c/\kappa^2)\delta_t$
  - use $\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})] = m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top = m\mathbf{U}^*\mathbf{B}^*\mathbf{B}^\top - m\mathbf{U}\mathbf{B}\mathbf{B}^\top$,
  - bound $\|\mathrm{diff}\|$ using sub-exponential Bernstein inequality and ϵ-net

N. Vaswani, "Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A Simple Proof", to appear IEEE Trans. Info Th., 2024

Proof outline: details I

Step 1: Bound initialization error using Wedin's $\sin\Theta$ theorem followed by sub-Gaussian Hoeffding inequality and a standard ϵ-net argument

- straightforward; difficult part was coming up with the truncation based initialization.

Step 2: **At iteration t, assume $\mathrm{SD}(\mathbf{U}^*, \mathbf{U}) \le \delta_t < \frac{0.1}{\sqrt{r}\kappa^2}$. Show $\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) \le (1 - c_0/\kappa^2)\delta_t$**

- Update of $\mathbf{B}$: $\mathbf{b}_k = (\mathbf{A}_k\mathbf{U})^\dagger \mathbf{y}_k$ for all $k \in [q]$
  - Use sub-expo Bernstein to show: w.p. at least $1 - q\exp(r - cm)$,

$$\|\mathbf{b}_k - \mathbf{U}^\top\mathbf{U}^*\mathbf{b}_k^*\| \le 0.4\|(\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{U}^*\mathbf{b}_k^*\| \text{ for all } k \in [q]$$

  - Use this to show
    - ★ Incoherence of $\mathbf{b}_k$'s, and
    - ★

$$\|\mathbf{B} - \mathbf{U}^\top\mathbf{U}^*\mathbf{B}^*\|_F^2 \le \sum_k \|(\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{U}^*\mathbf{b}_k^*\|^2 \le 0.16\delta_t^2\|\mathbf{B}^*\|_F^2$$

    - ★ $\|\mathbf{x}_k - \mathbf{x}_k^*\| \le 1.4\|(\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{U}^*\mathbf{b}_k^*\| \le 1.4\delta_t\|\mathbf{x}_k^*\|$
    - ★ Use above to show that $\|\mathbf{X} - \mathbf{X}^*\|_F \le 1.4\delta_t\|\mathbf{X}^*\|_F$
    - ★ Use above and Weyl to show: $\sigma_{\min}(\mathbf{B}) = \sigma_{\min}(\mathbf{X}) \ge 0.9\sigma_{\min}^*$ and $\sigma_{\max}(\mathbf{B}) = \sigma_{\max}(\mathbf{X}) \le 1.1\sigma_{\max}^*$

- Update of $\mathbf{U}$: $\tilde{\mathbf{U}}^+ = \mathbf{U} - \eta\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})$, $\mathbf{U}^+ = \mathrm{QR}(\tilde{\mathbf{U}}^+)$

Proof outline: details II

- Since $\mathbf{U}^+ = \tilde{\mathbf{U}}^+(\mathbf{R}^+)^{-1}$ and $\sigma_{\min}(\mathbf{R}^+) = \sigma_{\min}(\tilde{\mathbf{U}}^+)$, adding/subtracting $\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})]$, we get

$$\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) \leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})\tilde{\mathbf{U}}^+\|}{\sigma_{\min}(\tilde{\mathbf{U}}^+)} \leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})(\mathbf{U} - \eta\mathbb{E}[\nabla f(\mathbf{U}, \mathbf{B})]\| - \|\mathrm{diff}\|)}{1 - \eta\|\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})]\| - \eta\|\mathrm{diff}\|}$$

  where $\mathrm{diff} := \nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B}) - \mathbb{E}[\nabla_U f(\mathbf{U}, \mathbf{B})]$.

- Use
  - ★ $\mathbb{E}[\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{B})] = m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top = m\mathbf{U}^*\mathbf{B}^*\mathbf{B}^\top - m\mathbf{U}\mathbf{B}\mathbf{B}^\top$,
  - ★ $(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})\mathbf{U}^*\mathbf{B}^*\mathbf{B}^\top = \mathbf{0}$
  - ★ $\|\mathbb{E}[\nabla f(\mathbf{U}, \mathbf{B})]\| = m\|(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top\| \leq m\|\mathbf{X}^* - \mathbf{X}\|_F \sigma_{\max}(\mathbf{B}) \leq m1.5\delta_t\sqrt{r}{\sigma^*_{\max}}^2$
  - ★ $\|\mathrm{diff}\| \leq 1.1\delta_t{\sigma^*_{\min}}^2$ w.h.p. if $mq \geq C\kappa^4\mu^2 nr$:
    by applying sub-exponential Bernstein inequality and ϵ-net argument

Proof outline: details III

$$
\begin{aligned}
\mathrm{SD}(\mathbf{U}^*, \mathbf{U}^+) &\leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})\tilde{\mathbf{U}}^+\|}{\sigma_{\min}(\tilde{\mathbf{U}}^+)} \\
&\leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})(\mathbf{U} - \eta\mathbb{E}[\nabla f(\mathbf{U},\mathbf{B})] + \eta\mathbb{E}[\nabla f(\mathbf{U},\mathbf{B})] - \eta\nabla f(\mathbf{U},\mathbf{B}))\|)}{1 - \eta\|\nabla f(\mathbf{U},\mathbf{B})\|} \\
&\leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})(\mathbf{U} - \eta m\mathbf{U}\mathbf{B}\mathbf{B}^\top)\| + \|\mathrm{diff}\|)}{1 - \eta\|m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top\| - \eta\|\mathrm{diff}\|} \\
&\leq \frac{\|(\mathbf{I} - \mathbf{U}^*\mathbf{U}^{*\top})\mathbf{U}(\mathbf{I} - \eta m\mathbf{B}\mathbf{B}^\top)\| + \|\mathrm{diff}\|)}{1 - \eta\|m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top\| - \eta\|\mathrm{diff}\|} \\
&\leq \frac{\delta_t\|\mathbf{I} - \eta m\mathbf{B}\mathbf{B}^\top\| + \|\mathrm{diff}\|)}{1 - \eta\|m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top\| - \eta\|\mathrm{diff}\|} \\
&= \frac{\delta_t\lambda_{\max}(\mathbf{I} - \eta m\mathbf{B}\mathbf{B}^\top)\| - \|\mathrm{diff}\|)}{1 - \eta\|m(\mathbf{X}^* - \mathbf{X})\mathbf{B}^\top\| - \eta\|\mathrm{diff}\|} \quad \text{if } \eta \leq 0.5/m{\sigma^*_{\max}}^2 \\
&\leq \frac{\delta_t(1 - 0.9\eta m{\sigma^*_{\min}}^2) - \eta m c_1\delta_t{\sigma^*_{\min}}^2}{1 - \eta m 1.5\delta_t\sqrt{r}{\sigma^*_{\max}}^2 - \eta m c_1\delta_t{\sigma^*_{\min}}^2} \quad \text{whp if } mq \geq C\kappa^4\mu^2 nr \\
&\leq \delta_t(1 - c\eta m{\sigma^*_{\min}}^2) \quad \text{if } \delta_t \leq \frac{c_2}{\sqrt{r}\kappa^2} \\
&\leq \delta_t(1 - cc_\eta/\kappa^2) := \delta_{t+1} \quad \text{if } \eta = c_\eta/m{\sigma^*_{\max}}^2
\end{aligned}
$$

N. Vaswani, "Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A Simple Proof", to appear IEEE Trans. Info Th., 2024

Byzantine-Resilient AltGDmin

At most L_{byz} out of L nodes can be Byzantine with $L_{byz} < 0.5L - 1$.

- Geometric Median (GM) of gradients $\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_L$:

$$GM = \arg\min_{\mathbf{z}} \sum_{\ell=1}^{L} \|\mathbf{z} - \mathbf{z}_\ell\|_2$$

  Approx algorithms exist to compute the GM.

At most L_{byz} out of L nodes can be Byzantine with $L_{byz} < 0.5L - 1$.

- Geometric Median (GM) of gradients $\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_L$:

$$GM = \arg\min_{\mathbf{z}} \sum_{\ell=1}^{L} \|\mathbf{z} - \mathbf{z}_\ell\|_2$$

  Approx algorithms exist to compute the GM.

- Krum: find the $\mathbf{z}_\ell$ for which the sum of squared distances of the $(L - L_{byz} - 2)$ $\mathbf{z}_{\ell'}$'s closest to it is smallest.

Guarantees for Geometric Median [existing work]

- Given estimates $\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_L$ of an unknown vector $\mathbf{z}^*$. Suppose we can compute

$$\mathbf{z}_{GM} = \arg\min_{\mathbf{z}} \sum_{\ell=1}^{L} \|\mathbf{z} - \mathbf{z}_\ell\|_2$$

- If $(L - L_{byz})$ $\mathbf{z}_\ell$'s satisfy $\|\mathbf{z}_\ell - \mathbf{z}^*\|_2 \leq \epsilon \|\mathbf{z}^*\|_2$ then,

$$\|\mathbf{z}_{GM} - \mathbf{z}^*\|_2 \leq C\epsilon \|\mathbf{z}^*\|_2, \;\; C := C_{0.5-(L_{byz}/L)}$$

- Can extend above idea to high probability guarantee also.
- Cannot compute GM exactly; approx algorithms exist but then

Byzantine Resilient AltGDmin (Byz-AltGDmin) [3]

Recall

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots \mathbf{y}_k, \dots \mathbf{y}_q] := [\mathbf{A}_1\mathbf{x}_1^*, \mathbf{A}_2\mathbf{x}_2^*, \dots \mathbf{A}_k\mathbf{x}_k^*, \dots \mathbf{A}_q\mathbf{x}_q^*]$$

Let $\mathcal{S}_\ell$ be the set of columns k observed/sensed at node ℓ, i.e., node ℓ has access to

$$\{\mathbf{y}_k, \mathbf{A}_k\}, \ k \in \mathcal{S}_\ell$$

Byz-AltGDmin:

- **Initialization: not easy – next page**
- **Min step for B: done locally at the nodes, no change.**
- **GD step for U:**
  - **Node ℓ computes ∇_ℓ on its data and sends to center**
  - **Center computes $\nabla = GM(\nabla_\ell, \ell \in [L])$ and $\mathbf{U}^+ = QR(\mathbf{U} - \eta\nabla)$**
  - **Center broadcasts $\mathbf{U}^+$ to nodes**

[3] Singh, Vaswani, Byzantine Resilient and Fast Federated Few-Shot Learning, ICML 2024

Byzantine-resilient Spectral Initialization : Byz-res Federated PCA

Recall:

- Initialization involves computing r-SVD of matrix $\mathbf{X}_0$ defined earlier
- Federated setting: use power method for this computation

Byz-resilient modification:

- Option 1: share all data with center – communication inefficient
- Option 2: use GM in each power method iteration - communic-efficient but needs too many samples

Recall:

- Initialization involves computing r-SVD of matrix $\mathbf{X}_0$ defined earlier
- Federated setting: use power method for this computation

Byz-resilient modification:

- Option 1: share all data with center – communication inefficient
- Option 2: use GM in each power method iteration - communic-efficient but needs too many samples

Proposed solution: **Subspace (geometric) Median**

- Compute $(\mathbf{U}_0)_\ell$ at each node locally and then try to compute their "medians";
- But these are close only in SD and not in Frob norm;
- Fix – use the fact that $\mathrm{SD}_F(\mathbf{U}, \mathbf{U}^*) \approx \|\mathbf{U}\mathbf{U}^\top - \mathbf{U}^*\mathbf{U}^{*\top}\|_F$

Communication-efficient and sample-efficient solution

- Node ℓ computes $\mathbf{U}_\ell$ as top r singular vectors of $(\mathbf{X})_\ell$ and sends it to center
- Center computes $GM = mat(GM(vec(\mathbf{U}_\ell \mathbf{U}_\ell^\top), \ell \in [L]))$
- Center finds $\ell_{GM} = \arg\min_{\ell \in [L]} \|\mathbf{U}_\ell \mathbf{U}_\ell^\top - GM\|_F$
- Center outputs $\mathbf{U}_{\ell_{GM}}$, broadcasts to nodes

Theorem (Subspace Median)

Suppose at most $L_{byz} < 0.5L$ nodes are Byzantine. Center receives subspace estimates $\mathbf{U}_\ell$ computed by each node ℓ and computes their Subspace Median, $\mathbf{U}_{GM}$. If, for all the non-Byz nodes, $\mathrm{SD}(\mathbf{U}_\ell, \mathbf{U}^*) \leq \delta$ w.p. at least $1 - p$, then,

$$\mathrm{SD}(\mathbf{U}_{GM}, \mathbf{U}^*) \leq C_{0.5 - L_{byz}/L}\delta \text{ w.p. at least } 1 - \exp\left(-L\psi\left(0.5 - L_{Byz}/L, p\right)\right)$$

Theorem

Consider Byz-AltGDmin. Set $T = C\tilde{\kappa}^2 \log(1/\epsilon)$ *and* $\eta = 0.5/m\sigma^{*2}_{\max}$. *Assume*

1. *right singular vectors' incoherence;*
2. *at most* L_{byz} *nodes are Byzantine with* $L_{byz} < 0.4L$; *and*
3. $\max_{\ell \neq \ell'} \|\mathbf{B}^*{}_\ell - \mathbf{B}^*{}_{\ell'}\|_F \leq G_B \sigma^*_{\max}$ *with* $G_B \leq \frac{c}{\tilde{\kappa}^2}$

If

$$mq \geq L\tilde{\kappa}^{10}\mu^2 nr(r + \log(1/\epsilon)),$$

and $m \geq \tilde{\kappa}^6 \max(\log q, r) \log\left(\frac{1}{\epsilon}\right)$, *then w.p. at least* $1 - 2Ln^{-10} - C\tilde{\kappa}^2 \log(1/\epsilon)Ln^{-10}$

$$\mathrm{SD}_F(\mathbf{U}^*, \mathbf{U}) \leq \max(\epsilon, 14G_B)$$

Novel ideas: (i) Subspace Median approach and its analysis; (ii) careful application of GM guarantee for the gradient

[4] Singh, Vaswani, Byzantine Resilient and Fast Federated Few-Shot Learning, ICML 2024

AltGDmin for other partly-decoupled LR problems

LRPR: recover $\mathbf{X}^*$ from $\mathbf{z}_k := |\mathbf{A}_k \mathbf{x}_k^*|, k = 1, 2, \ldots q$

$$\min_{\mathbf{U},\mathbf{B}} \sum_k \| \mathbf{z}_k - |\mathbf{A}_k \mathbf{U} \mathbf{b}_k| \|_2^2 = \min_{\mathbf{U},\mathbf{B},\mathbf{c}_k} \sum_k \| \mathbf{z}_k - \mathrm{diag}(\mathbf{c}_k)(\mathbf{A}_k \mathbf{U} \mathbf{b}_k)\|_2^2$$

where $\mathbf{c}_k$: vector of phases of $\mathbf{A}_k \mathbf{U} \mathbf{b}_k$

- Algorithm: (i) different init; (ii) recover $\mathbf{b}_k, \mathbf{c}_k$ by PR (iii) replace $\mathbf{y}_k$ by $\hat{\mathbf{y}}_k = \hat{\mathbf{c}}_k \circ \mathbf{z}_k$ in GD step
- Guarantee: Need $mq \gtrsim nr^2 \max(r, \log(1/\epsilon))$ samples[5]
- AltGDmin is most communication efficient. Once again the commonly studied GD methods do not apply.
- Novelty:
  - need to bound one extra error term because $\mathbf{y}_k$ by $\hat{\mathbf{y}}_k$ in gradient – use bounds from our past work on AltMin for LRPR.

[5] Nayer & Vaswani, Fast and Sample-Efficient Federated Low Rank Matrix Recovery from column-wise Linear and Quadratic Projections, IEEE Trans. Info. Th., 2023]

AltGDmin and Byz-AltGDmin for LR matrix completion (LRMC)

Recover $\mathbf{X}^*$ from a subset of its entries, $\mathbf{Y} := (\mathbf{X}^*)_\Omega$; i.i.d. Bernoulli observed entries

$$\min_{\mathbf{U},\mathbf{B}} \sum_k \|\mathbf{y}_k - \mathcal{P}_{\Omega_k}(\mathbf{U}\mathbf{b}_k)\|_2^2$$

- Algorithm: (i) different init; (ii) rest of AltGDmin: same as for LRCS
- Guarantee: Need $nr^2 \log n \log(1/\epsilon)$ samples on average[6]
- AltGDmin is most communication-efficient. Compute cost as good as that of AltMin or Factorized-GD.
- Novelty:
  - main challenge: need to show incoherence of $\mathbf{U}$ at each iteration
  - proof needs to use matrix Bernstein;
  - Byz-AltGDmin: need guaranteed incoherence of $\mathbf{U}$ at each iteration: need to carefully modify the Byz algorithm to ensure this – filter out bad gradients[7]

[6] Abbasi & Vaswani, Efficient Federated Low Rank Matrix Completion, IEEE Trans. Info. Th., 2025

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

Rpbust PCA: Recover LR $\mathbf{X}^*$ from $\mathbf{Y} := \mathbf{X}^* + \mathbf{S}^*$

- Ongoing work – submitted
- Much more difficult: there are three blocks now $\mathbf{U}, \mathbf{B}, \mathbf{S}$

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

- The naive approach: use $\mathbf{Z}_{\text{slow}} = \mathbf{U}$, $\mathbf{Z}_{\text{fast}} = \{\mathbf{B}, \mathbf{S}\}$ fails
- Need a carefully designed multi-block extension of AltGDmin
- New proof ideas needed to bound the sparse component and show incoherence of $\mathbf{U}$

AltGDmin-MRI: AltGDmin-LRCS for Dynamic MRI

- MRI problem: Recover an $n \times q$ matrix $\mathbf{Z}^* = [\mathbf{z}_1^*, \mathbf{z}_2^*, \ldots, \mathbf{z}_k^*, \ldots \mathbf{z}_q^*]$ from its undersampled fourier measurements,

$$\mathbf{y}_k = \mathbf{A}_k \mathbf{z}_k^*, \quad k = 1, 2, \cdots, q$$

where

  - $\mathbf{y}_k$ is an m_k length vector,
  - $\mathbf{A}_k = \mathbf{H}_k \mathbf{F}$, where $\mathbf{A}_k$ is an $m_k \times n$ matrix with $m_k < n$,
    - $\mathbf{F}$ is an $n \times n$ matrix that models 2D-DFT.
    - $\mathbf{H}_k$ is an $m_k \times n$ row selection matrix – models undersampling in Fourier domain

- Image sequence:
  - tensor of size $n_x \times n_y \times q$; we flatten it to an $n \times q$ matrix with $n = n_x n_y$. Thus $\mathbf{z}_k^*$ is k-th vectorized image

- Model the image sequence as being approximately LR after mean image subtraction – changes within the sequence are only governed by a few (r) factors, i.e.,

$$\mathbf{z}_k^* = \bar{\mathbf{z}}^* + \mathbf{x}_k^* + \mathbf{e}_k^*, \ k \in [q]$$

$\bar{\mathbf{z}}^*$: mean image, $\mathbf{e}_k^*$: small modeling error in this model

AltGDmin-MRI algorithm[8]

- Very similar images: first approximate and subtract out a "mean image"
- Only approximately LR: column-wise model-error correction step after altGDmin

Algorithm 1 AltGDmin-MRI

1: $\hat{\bar{\mathbf{z}}} \leftarrow \min_{\bar{\mathbf{z}}} \sum_{k=1}^{q} \|\mathbf{y}_k - \mathbf{A}_k \bar{\mathbf{z}}\|^2$

2: Compute $\tilde{\mathbf{y}}_k \leftarrow \mathbf{y}_k - \mathbf{A}_k \hat{\bar{\mathbf{z}}}$ for each $k = 1, 2, \cdots, q$ and use these as input for basic AltGDmin. Denote its output as $\hat{\mathbf{X}}$

3: For each $k \in q$, compute $\tilde{\tilde{\mathbf{y}}}_k = \mathbf{y}_k - \mathbf{A}_k \hat{\bar{\mathbf{z}}} - \mathbf{A}_k \hat{\mathbf{x}}$ and run 3 iterations of GD to solve $\min_e \|\tilde{\tilde{\mathbf{y}}}_k - \mathbf{A}_k \mathbf{e}\|^2$. Denote its output by $\hat{\mathbf{e}}_k$.

4: **Output:** $\hat{\mathbf{Z}} = [(\mathbf{z}_1)_T, (\mathbf{z}_2)_T, \cdots (\mathbf{z}_q)_T]$ with $\hat{\mathbf{z}}_k = \hat{\bar{\mathbf{z}}} + \hat{\mathbf{x}}_k + \hat{\mathbf{e}}_k$

Without parameter-tuning, AltGDmin-MRI gives the (near-)best reconstructions for 25 datasets – different applications, sampling schemes and rates
Is also one of the fastest.
Beats Deep Learning – need application-specific architecture and parameter tuning

[8] Fast Low Rank Compressive Sensing for Accelerated Dynamic MRI, IEEE Trans. Comput Imaging, 2023.

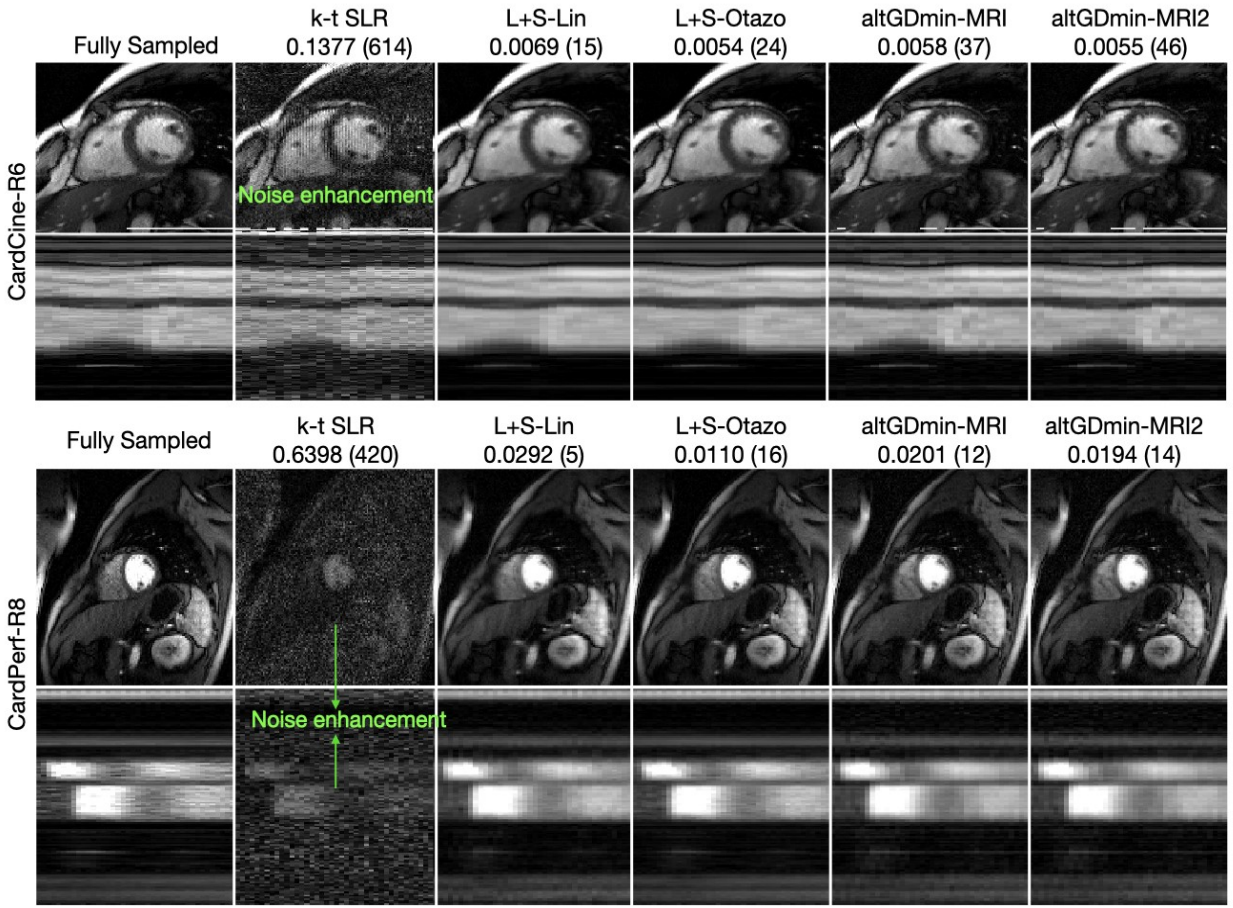
Results: Comparison of proposed algorithm with kt-SLR, L+S-Otazo, L+S-Lin

Dataset	kt-SLR	L+S-Otazo	L+S-Lin	altGDmin-MRI1	altGDmin-MRI2
Cartesian					
CardPerf-R8	0.6398 (420.71)	0.0110 (16.93)	0.0292 (5.99)	0.0201 (12.01)	0.0194 (14.74)
CardCine-R6	0.1377 (614.53)	0.0054 (24.57)	0.0069 (15.51)	0.0058 (37.55)	0.0055 (46.08)
Pseudo-radial					
Brain(4)	0.0093 (127.92)	0.0167 (5.65)	0.0173 (2.53)	0.0125 (3.70)	0.0121 (4.86)
Brain(8)	0.0034 (102.33)	0.0086 (4.15)	0.0095 (2.48)	0.0054 (3.87)	0.0051 (5.11)
Brain(16)	0.0014 (75.57)	0.0049 (2.81)	0.0062 (2.44)	0.0027 (3.84)	0.0024 (4.98)
Speech(4)	0.1543 (5234.83)	0.1991 (537.91)	0.2545 (304.50)	0.1416 (131.87)	0.1395 (153.43)
Speech(8)	0.0593 (5261.49)	0.1107 (491.11)	0.1284 (306.16)	0.0991 (152.35)	0.0952 (176.35)
Speech(16)	0.0203 (5288.61)	0.0550 (426.10)	0.0557 (304.45)	0.0580 (236.85)	0.0540 (261.39)
UnCardPerf(4)	0.0894 (4150.72)	0.0910 (189.88)	0.1424 (50.68)	0.0695 (70.97)	0.0684 (92.31)
UnCardPerf(8)	0.0442 (3472.96)	0.0591 (120.55)	0.0632 (50.44)	0.0470 (67.79)	0.0451 (90.66)
UnCardPerf(16)	0.0206 (2873.30)	0.0370 (88.44)	0.0329 (50.47)	0.0298 (69.08)	0.0275 (90.16)
CardOCMR16(4)	0.0362 (227.92)	0.0293 (10.73)	0.0515 (2.75)	0.0092 (10.52)	0.0092 (15.42)
CardOCMR16(8)	0.0045 (225.98)	0.0064 (8.37)	0.0101 (2.73)	0.0033 (7.26)	0.0033 (8.64)
CardOCMR16(16)	0.0015 (162.12)	0.0035 (4.64)	0.0030 (2.73)	0.0015 (5.39)	0.0014 (6.69)
CardOCMR19(4)	0.0216 (399.70)	0.0251 (18.70)	0.0698 (5.06)	0.0095 (11.58)	0.0094 (14.10)
CardOCMR19(8)	0.0043 (409.01)	0.0092 (13.05)	0.0149 (5.06)	0.0051 (10.49)	0.0050 (12.91)
CardOCMR19(16)	0.0020 (269.01)	0.0052 (7.20)	0.0044 (5.05)	0.0032 (9.47)	0.0030 (12.00)
PINCAT(4)	0.0445 (34.85)	0.0381 (8.04)	0.1054 (2.26)	0.0278 (1.63)	0.0278 (1.77)
PINCAT(8)	0.0216 (31.54)	0.0162 (3.91)	0.0208 (2.22)	0.0166 (1.36)	0.0166 (1.31)
PINCAT(16)	0.0095 (23.31)	0.0065 (2.70)	0.0047 (2.25)	0.0097 (1.08)	0.0097 (1.13)
avg-Err (avg-Time)	0.0663 (1470.3)	0.0369 (99.3)	0.0515 (56.3)	0.0289 (42.4)	0.0280 (50.7)

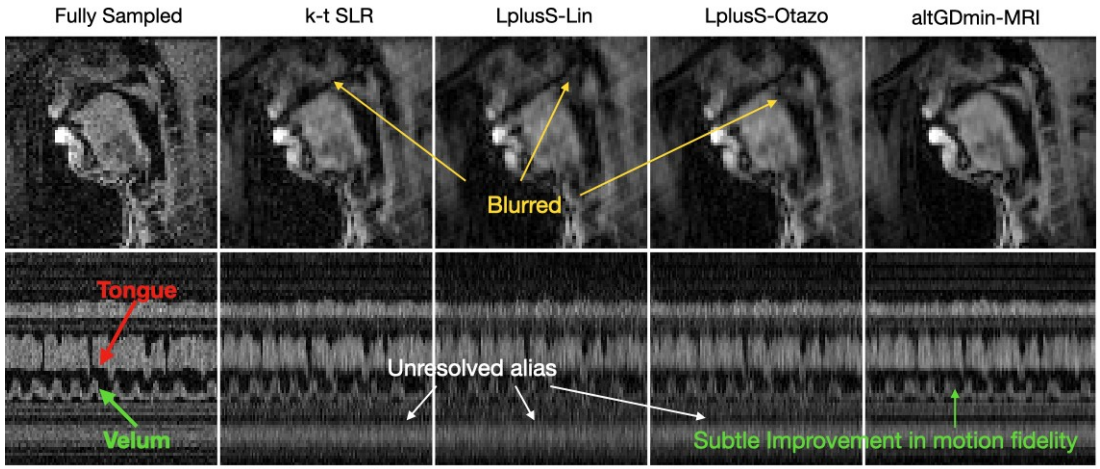
Table 1: *Table format is* **Error (Recon time in seconds)**. *The last row shows* **average-Error (average-Reconstruction time in seconds)** *over all 20 rows of results.*

All experiments for all apps used the SAME set of algorithm parameters. Also visually evaluated on real MRI scanner data

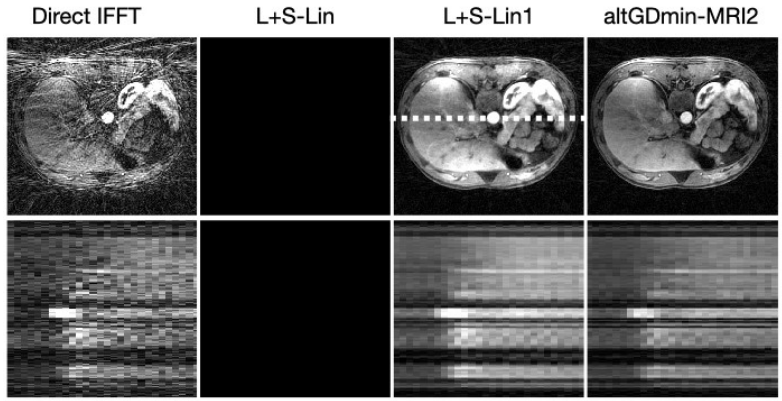
Visual Comparisons:



Retrospective, Cartesian, Cardiac Perfusion and Cardiac Cine sequences: Comparisons of reconstruction algorithms on CardPerf-R8 and CardCine-R6 datasets. In row 1, we show one original frame (14th frame) and its reconstructions. In row 2, we show the corresponding time profile images.



Retrospective Pseudo-radially sampled Vocal Tract while Speaking. In row 1, we show one original frame and its reconstructions. In row 2, we show the corresponding time profile images.



Prospective (real MRI scanner data) Radial-sampled Abdomen dataset. L+S-Lin fails when using same parameters as those for other expts; it works when using tuned parameters (L+S-Lin1)

Open Questions

Always looking for interested Ph.D. students or postdocs

- AltGDmin for other Partly-Decoupled Optimization Problems
  - Shuffled / Permuted LR Column-wise Sensing
  - Tensor extensions
- AltGDmin for optimizing a general $f(\mathbf{Z}_{\rm slow}, \mathbf{Z}_{\rm fast})$:
  - Assume min w.r.t. $\mathbf{Z}_{\rm fast}$ is decoupled and fast, and can be solved exactly.
  - Suppose also we assume local strong convexity and smoothness w.r.t. $\mathbf{Z}_{\rm slow}$, for every value of $\mathbf{Z}_{\rm fast}$ in the neighborhood of the initialization
  - What can we say about AltGDmin? What about its SGD extension?

References:
Byzantine-Resilient Federated Alternating Gradient Descent and Minimization for Partly-decoupled Low Rank Matrix Learning, ICML, 2025.
Byzantine Resilient and Fast Federated Few-Shot Learning, ICML, 2024.
Byzantine-Resilient Federated PCA and Low Rank Column-wise Sensing, IEEE Trans Info Theory, 2024.

Efficient Federated Low Rank Matrix Completion, IEEE Trans Info Theory, 2025.
Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A Simple Proof, IEEE Trans Info Theory, 2024.
Fast Sample-Efficient Federated Low Rank Matrix Recovery from Column-wise Linear and Quadratic Projections, IEEE Trans Info Theory, 2023

Fast Low Rank Compressive Sensing for Accelerated Dynamic MRI, IEEE Trans. Computational Imaging, 2023.

Early Math for Comm/SP/IT/STEM success

Why fix math early?

1. Math is Cumulative
   - Arithmetic needed to understand basic algebra; algebra needed for linear algebra, probability, calculus.
2. Workforce Success - Engineers designing safe bridges, planes, flying software
   - We would like to teach ML or SP to well-prepared students
   - Industry would like well-educated engineers
   - Even careful use of AI for code writing, deploying, combining needs an understanding of what problem the code is solving, when it would fail to return accurate solutions, e.g., Least Squares estimation
   - Renewed need to understand algorithmic aspects of coding, not as much syntax
3. Learning and Earning Equity:
   - For those w/o good early math skills, later math becomes difficult to understand; good math skills lead to STEM careers – better average pay
4. Low-Cost – least expensive to teach or tutor but high payoff
   - No science or engineering kits needed, no labs needed
   - May not work for all, but will for say one in five.
   - Most engineering just needs good math, not genius-level

What can we do to help fix math for all?

1. Math tutoring and support programs
   - Through our university, company, public library, community group, schools run it and we staff it (ideal)
2. Awareness of the need and resources for building good early math skills
   - Use of social/news media - by us or get our org to do it (univ, company, IEEE)
   - Why learn 5th grade math – cumulative, earning potential, not crash planes
   - How - Khan Academy or other apps, textbooks, workbooks/worksheets
3. Advocate for math education policies in local schools w/ long-term success in mind
   - School policies are often influenced by new Education research
   - Due to many practical limitations, this research is usually short-term 2-3 years
   - E.g.: no one considers impact of removing homework from 5th grade on 12th or college math success

CyMath-ISU K-12 Math Support

- Hybrid mode all year weekly math tutoring – after-school at 2 schools, ISU, Zoom
  - Supports 3rd-12th graders; those who start younger tend to stay on
  - At-home math practice resources – ALEKS or Khan Academy – and encouragement
- Tutors
  - STEM grad student or faculty volunteers;
  - and 1-3 program assistants (undergrads - tutor & manage logistics) per site
- How - no one model, resource, group-size
  - generally 1 tutor for 1-3 students; all others in 1-2 large groups with paid staff
  - use of app: simplifies tutoring, no prep needed, build lesson off of app questions, encourage practice
  - almost no tutor training, just observe sessions or have others observe you
  - use texting with parents - encourage at-home practice
- History and Stats:
  - Started in Fall 2020; re-started in 2023 in hybrid mode.
  - 100+ enrolled students, 60+ tutors, 8-10 paid program assistants

Impact I

- CyMath now supports 100 students in grades 3-11 (not all come).
- Summer 2025 info: Of the 15 3rd-6th graders who spent at least a year at CyMath:
  - All grew; a third (6/15) showed very large growth of more than 20-percentiles and stayed at the higher level
    - ⋆ AM (grade 3, Fall'23) went up from 46th to 85th percentile in a year and has remained at that level or gone higher
    - ⋆ NW (grade 3, Fall'23) went from 20th to 55th percentile in a year and has remained at that level or higher
    - ⋆ GO (grade 5, Fall'24) went from 34th to 80th percentile in a year
    - ⋆ MA (grade 3, Fall'24) went from 78th to 98th percentile in a year
    - ⋆ TM (grade 4, LateSpring'24) went from 34th to 70th percentile in a year
    - ⋆ NH (grade 5, LateSpring'24) went from 7th to 28th percentile in a year
  - Many others moved up about 10-15 percentile points; but adaptive testing is very noisy, so we only mention large and sustained changes
- First survey of all tutors conducted in Summer 2025: 18 respondents
  - 94% said tutoring helped improve their teaching skills,
  - 83% said it improved their communication skills, and
  - 44% said it also helped them find community

Impact II

- Grad students – find extra mentors, reference letter stronger than just "took my class and did well"
- Education majors (future teachers):
  - Tutoring alongside STEM folks hopefully impacts their future teaching, understand what's critical for future K-12 to college pipeline
- Other impact on faculty: grant proposals – math outreach greatly appreciated; NSF CAREER – 2 letters, 1 got; faculty parents – learn about US schools or how to help our own children, or nag them less.

Open Questions I

- Should K-5/K-8 policies be designed w/ STEM success "likelihood" in mind?
  - Education research studies are usually short-term: 2-3 years long
    - ⋆ Should STEM or HS educator perspectives be also considered instead of only relying on research for policymaking?
  - Example: homework and studying-for-a-test skills – needed for college success
- Making math fun or first real math then fun?
  - Is it better to first just do "real math" and then allow "fun" time
  - Our experience suggests latter is easier, cheaper, scalable, saves time – even kids' focus-time is very limited
- University/Company Extension/Outreach – use Social and News Media
  - Provide math awareness – need & resources for early math skills. See CyMath math-for-all page
- Community groups or youth programs free or paid or religious groups
  - Add 30-minutes of math to each youth program
  - Run math tutoring programs
- In-school math tutoring – reaches all kids and known to be most effective

- Older students within the school help younger ones
- Recruit work-from-home STEM people, parents, STEM retirees, school administrators

- High Schoolers Math++ Teach and Learn
  - Train HS students to tutor K-8 students in their own district after-school
  - May be doable if a math teacher can be a (paid) supervisor and coordinator for it HS students receive math++ tutoring/mentoring by ISU students