Taming Heterogeneity in Federated Linear Stochastic Approximation

Paul Mangold
CMAP, École Polytechnique, France

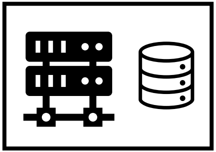
Joint work with
S. Samsonov, S. Labbi, I. Levin, R. Alami, A. Naumov, E. Moulines

September 5, 2024

Background on Federated Learning

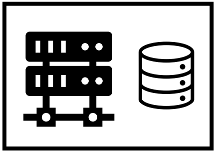
Data Collection

Data center



Data Collection

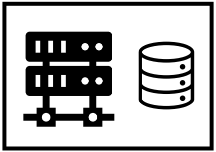
Data center



vs.

Data Collection

Data center

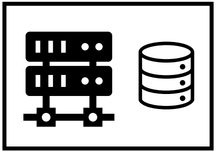


vs.

Data collection *by users*

Data Collection

Data center



vs.

Data collection *by users*

→ **how to use all this data?**

Centralizing in a data center is difficult

Centralizing data is often impossible

- *Privacy*:
  $\rightarrow$ data may be sensitive (e.g. health records, geolocation)
- *Volume of data*:
  $\rightarrow$ data may be large (e.g. cameras of self-driving car)
- *Time*:
  $\rightarrow$ it may be needed to take decisions quickly (e.g. reinforcement learning)

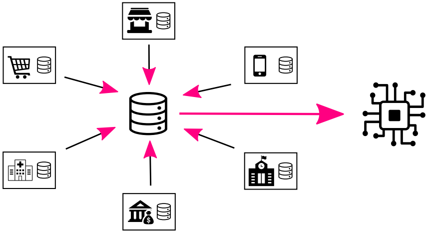
Why share in the first place?

If it is so difficult to share data... why do it?

- local datasets are often too small
  $\rightarrow$ no statistical significance (e.g. medical study)

- local datasets can be biased
  $\rightarrow$ if a self-driving car learns in countryside, can it drive in the city?

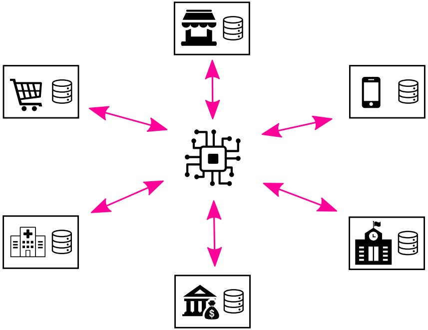
Classical vs Federated Learning



A single optimization problem

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x,y \sim D} \left[\ell(\theta; x, y) \right]$$

Classical vs Federated Learning



Multiple sub-problems

$$\min_{\theta \in \mathbb{R}^d} \sum_{c=1}^{N} \mathbb{E}_{x^c, y^c \sim D^c} \left[\ell(\theta; x^c, y^c) \right]$$

$\rightarrow$ but only *one shared solution*

Best Scenario: Homogeneous Data

N local sub-problems

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^1, y^1 \sim D^1} \left[\ell(\theta; x^1, y^1) \right] \to \theta_\star^1$$

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^2, y^2 \sim D^2} \left[\ell(\theta; x^2, y^2) \right] \to \theta_\star^2$$

$$\vdots$$

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^N, y^N \sim D^N} \left[\ell(\theta; x^N, y^N) \right] \to \theta_\star^N$$

Best Scenario: Homogeneous Data

N local sub-problems

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^1, y^1 \sim D^1} \left[\ell(\theta; x^1, y^1) \right] \rightarrow \theta_\star^1$$

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^2, y^2 \sim D^2} \left[\ell(\theta; x^2, y^2) \right] \rightarrow \theta_\star^2$$

$$\vdots$$

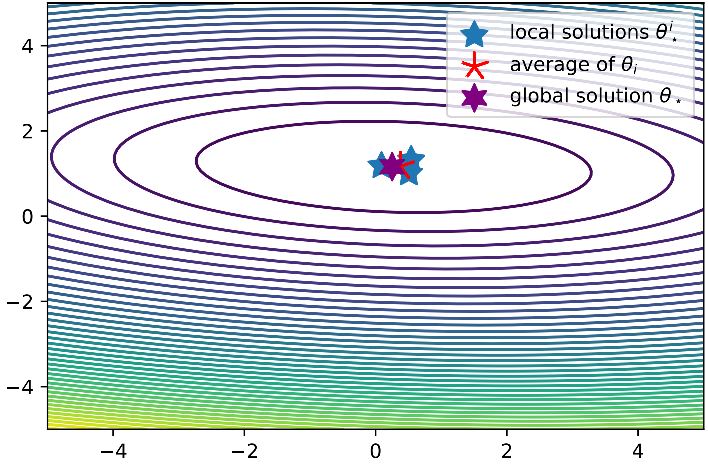
$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{x^N, y^N \sim D^N} \left[\ell(\theta; x^N, y^N) \right] \rightarrow \theta_\star^N$$

Estimate global solution

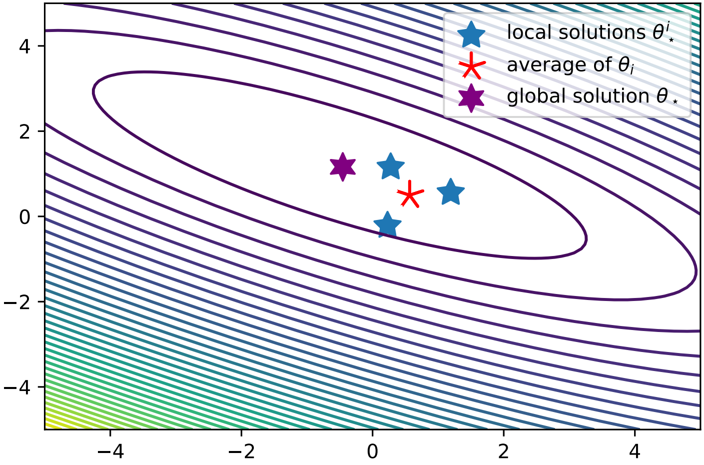
$$\theta_\star = \frac{1}{N} \sum_{c=1}^{N} \theta_\star^c$$

OK if $\mathcal{D}_1 = \mathcal{D}_2 = \cdots = \mathcal{D}_N$

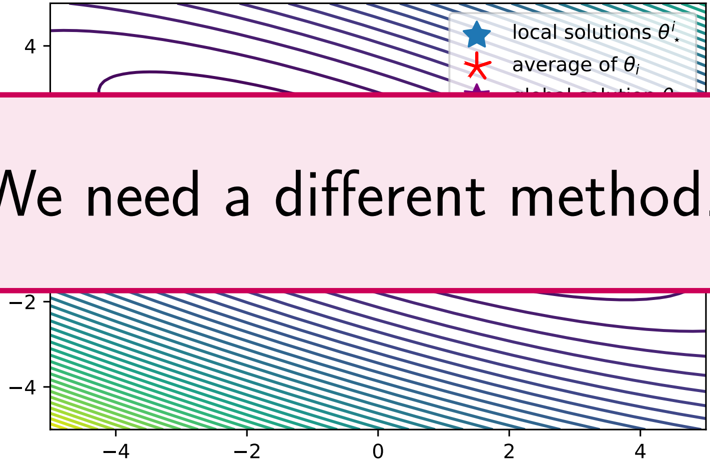
Best Scenario: Homogeneous Data



Failure: Heterogeneous Data



Failure: Heterogeneous Data



We need a different method...

Federated Optimization

$$\theta_\star \in \arg\min_{\theta \in \mathbb{R}^d} \sum_{c=1}^{N} f^c(\theta) \;, \qquad \text{where } f^c(\theta) = \mathbb{E}_{x^c, y^c \sim D^c}\left[\ell(\theta; x^c, y^c)\right]$$

[1] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *AISTATS*. PMLR. 2017, pp. 1273–1282.

Federated Optimization

$$\theta_\star \in \arg\min_{\theta\in\mathbb{R}^d} \sum_{c=1}^{N} f^c(\theta) \ , \qquad \text{where } f^c(\theta) = \mathbb{E}_{x^c,y^c\sim D^c}\left[\ell(\theta; x^c, y^c)\right]$$

Federated Averaging (or local (S)GD)[1]

- For each $t = 0\ldots$:
  - Set $\theta^c_{t,0} = \theta_t$
  - For each agent c, do H gradient updates:

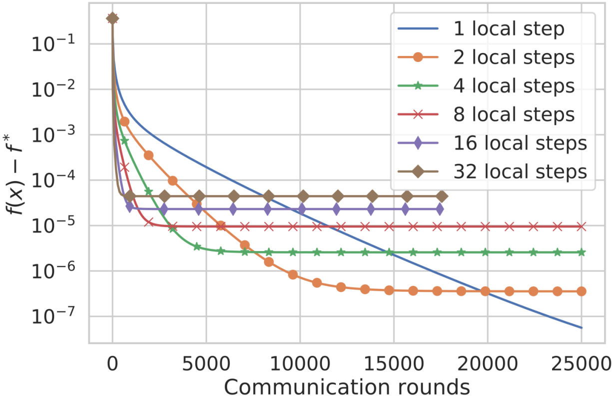
$$\theta^c_{t,h+1} = \theta^c_{t,h} - \eta\nabla f^c(\theta^c_{t,h})$$

- Aggregate models: $\theta_{t+1} = \frac{1}{N}\sum_{c=1}^{N}\theta^c_{t,H}$

[1] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *AISTATS*. PMLR. 2017, pp. 1273–1282.

Communication and Sample Complexity

Local Training vs. Precision



(Figure from Ahmed Khaled, Konstantin Mishchenko, and Peter Richtarik. "Tighter Theory for Local SGD on Identical and Heterogeneous Data". In: *AISTATS*. 2020, pp. 4519–4529)

Beyond Federated Optimization: Federated TD and LSA

Some problems do not fit this framework...

Example: TD Learning with linear approximation (I)

In Federated TD learning, N agent use a shared policy π in N different environments:

$$S_0^c = s, A_k^c \sim \pi(\cdot|S_k^c), \text{ and } S_{k+1}^c \sim P_{\mathrm{MDP}}^c(\cdot|S_k^c, A_k^c)$$

Some problems do not fit this framework...

Example: TD Learning with linear approximation (I)

In Federated TD learning, N agent use a shared policy π in N different environments:

$$S_0^c = s, A_k^c \sim \pi(\cdot|S_k^c), \text{ and } S_{k+1}^c \sim P_{\text{MDP}}^c(\cdot|S_k^c, A_k^c)$$

Goal: estimate its value in each environment, for $s \in \mathcal{S}$,

$$V^{c,\pi}(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r^c(S_k^c, A_k^c)\right]$$

where r^c is a reward obtained by agent c

Some problems do not fit this framework...

Example: TD Learning with linear approximation (II)

Idea: build a *shared estimate* of all values

$$V^{c,\pi}(s) \approx \theta^\top \varphi(s)$$

using $\theta \in \mathbb{R}^d$ and embedding $\varphi : \mathcal{S} \to \mathbb{R}^d$

Some problems do not fit this framework...

Example: TD Learning with linear approximation (II)

Idea: build a *shared estimate* of all values

$$V^{c,\pi}(s) \approx \theta^\top \varphi(s)$$

using $\theta \in \mathbb{R}^d$ and embedding $\varphi : \mathcal{S} \to \mathbb{R}^d$

Is this meaningful to use a shared estimate? Yes, because:

- If agents are homogeneous, it reduces sample complexity
- If agents are heterogeneous, it may reduce bias of local data

Linear Stochastic Approximation

Special case: only one agent

TD (with linear approx.) can be seen as solving a linear system

$$A\theta_\star = b$$

where A and b are known through stochastic estimates $A(Z)$, $b(Z)$

Linear Stochastic Approximation

Special case: only one agent

TD (with linear approx.) can be seen as solving a linear system

$$A\theta_\star = b$$

where A and b are known through stochastic estimates $A(Z)$, $b(Z)$

Note: It is inefficient to cast it as a minimization problem
$\rightarrow$ This requires a different method, with a different analysis

Algorithm for LSA

Initialize $\theta_0 \in \mathbb{R}^d$
for $t = 0$ to $T - 1$ **do**
Observe Z_t and update:

$$\theta_t = \theta_{t-1} - \eta(A(Z_t)\theta_{t-1} - b(Z_t))$$

end for

Context, idea on nice analysis (I)[2]

Initialize $\theta_0 \in \mathbb{R}^d$
for $t = 0$ to $T - 1$ **do**
  Observe $Z^c_{t,h}$ and update: $\theta_t = \theta_{t-1} - \eta(A(Z_t)\theta_{t-1} - b(Z_t))$
end for

[2] Sergey Samsonov et al. "Improved High-Probability Bounds for the Temporal Difference Learning Algorithm via Exponential Stability". In: *COLT*. PMLR. 2024, pp. 4511–4547.

Context, idea on nice analysis (I)[2]

Initialize $\theta_0 \in \mathbb{R}^d$
for $t = 0$ to $T - 1$ **do**
  Observe $Z_{t,h}^c$ and update: $\theta_t = \theta_{t-1} - \eta(A(Z_t)\theta_{t-1} - b(Z_t))$
end for

Stochastic Expansion

We may write: $\theta_t - \theta_\star = (\mathrm{Id} - \eta A(Z_t))(\theta_{t-1} - \theta_\star) - \eta\varepsilon(Z_t)$

[2] Sergey Samsonov et al. “Improved High-Probability Bounds for the Temporal Difference Learning Algorithm via Exponential Stability”. In: *COLT*. PMLR. 2024, pp. 4511–4547.

Context, idea on nice analysis (I)[2]

Initialize $\theta_0 \in \mathbb{R}^d$
for $t = 0$ to $T-1$ **do**
  Observe $Z^c_{t,h}$ and update: $\theta_t = \theta_{t-1} - \eta(A(Z_t)\theta_{t-1} - b(Z_t))$
end for

Stochastic Expansion

We may write: $\theta_t - \theta_\star = (\mathrm{Id} - \eta A(Z_t))(\theta_{t-1} - \theta_\star) - \eta\varepsilon(Z_t)$

Assumptions

- Oracle: i.i.d sequence Z_t's such that $\mathbb{E}[A(Z_t)] = A$, and $\mathbb{E}[b(Z_t)] = b$
- Exponential stability: $\mathbb{E}[\| \prod_{t=\ell}^{k}(\mathrm{Id} - \eta A(Z_t))\|^2] \leq (1-\eta a)^{k-\ell}$ for some $a > 0$
- Noise $\varepsilon(Z) = (A(Z) - A)\theta_\star + (b(Z) - b)$ has finite variance $\sigma_\star^2$

[2] Sergey Samsonov et al. "Improved High-Probability Bounds for the Temporal Difference Learning Algorithm via Exponential Stability". In: *COLT*. PMLR. 2024, pp. 4511–4547.

Context, idea on nice analysis (II)[3]

Stochastic Expansion

$$\theta_T - \theta_\star = \Gamma_{1:T}(\theta_0 - \theta_\star) + \eta \sum_{t=1}^{T} \Gamma_{t+1:T}\varepsilon(Z_t)$$

Where $\Gamma_{t:t'}$ "accumulates the updates" from t to t':

$$\Gamma_{t:t'} = (\mathrm{Id} - \eta A(Z_{t'}))(\mathrm{Id} - \eta A(Z_{t'-1})) \cdots (\mathrm{Id} - \eta A(Z_t))$$

[3] Sergey Samsonov et al. "Improved High-Probability Bounds for the Temporal Difference Learning Algorithm via Exponential Stability". In: *COLT*. PMLR. 2024, pp. 4511–4547.

Context, idea on nice analysis (III)[4]

Stochastic Expansion

$$\theta_T - \theta_\star = \Gamma_{1:T}(\theta_0 - \theta_\star) + \eta \sum_{t=1}^{T} \Gamma_{t+1:T}\varepsilon(Z_t)$$

Using $\mathbb{E}[\|\Gamma_{t:t'}u\|^2] \leq (1-\eta a)^{t'-t+1}\|u\|^2$ to bound each term:

$$\mathbb{E}[\|\theta_T - \theta_\star\|^2] \leq (1-\eta a)^T \|\theta_0 - \theta_\star\|^2 + \frac{\eta\sigma_\star^2}{a}$$

[4] Sergey Samsonov et al. "Improved High-Probability Bounds for the Temporal Difference Learning Algorithm via Exponential Stability". In: *COLT*. PMLR. 2024, pp. 4511–4547.

Federated LSA

Take A^c, b^c such that $A^c\theta_\star^c = b^c$ for $c = 1..N$

Federated LSA

Take A^c, b^c such that $A^c\theta_\star^c = b^c$ for $c = 1..N$

Goal: solve collaboratively

$$\left(\frac{1}{N}\sum_{c=1}^{N} A^c\right)\theta_\star = \frac{1}{N}\sum_{c=1}^{N} b^c$$

Assumptions

- $\theta_\star$ and $\theta_\star^c$ are unique, and A^c and b^c are split among N agents
- Oracle: i.i.d sequence Z_t^c's such that $\mathbb{E}[A(Z_t^c)] = A^c$, and $\mathbb{E}[b(Z_t^c)] = b^c$
- Exponential stability: $\mathbb{E}[\| \prod_{t=\ell}^{k}(\mathrm{Id} - \eta A^c(Z_t^c))\|^2] \leq (1-\eta a)^{k-\ell}$ for $a > 0$
- Noise $\varepsilon^c(Z) = (A^c(Z) - A^c)\theta_\star^c + (b^c(Z) - b^c)$ has variance bounded by $\sigma_\star^2$

Solving Federated LSA

FedLSA Algorithm

for $t = 0$ to $T - 1$ **do**

Initialize $\theta_{t,0} = \theta_t$

for each agent $c = 1..N$ **do**

for $h = 1$ to H **do**

Observe $Z^c_{t,h}$ and perform local update:

$$\theta_{t,h} = \theta^c_{t,h-1} - \eta(A^c(Z^c_{t,h})\theta^c_{t,h-1} - b^c(Z^c_{t,h}))$$

end for

end for

Aggregate local updates $\theta_{t+1} = \frac{1}{N}\sum_{c=1}^{N} \theta^c_{t,H}$

end for

Analysis of FedLSA

Stochastic Expansion (over one communication round)

$$\theta_t - \theta_\star = \frac{1}{N}\sum_{c=1}^{N} \Gamma^c_{t,1:H}(\theta_{t-1} - \theta_\star) + \eta \sum_{c=1}^{N}(\mathrm{Id} - \Gamma^c_{t,1:H})(\theta^c_\star - \theta_\star)$$

$$+ \frac{\eta}{N}\sum_{c=1}^{N}\sum_{h=1}^{H} \Gamma^c_{t,h+1:H}\varepsilon^c(Z^c_t)$$

Where $\Gamma^c_{t,h:h'}$ "accumulates local updates", round t, from h to h',

$$\Gamma^c_{t,h:h'} = (\mathrm{Id} - \eta A^c(Z^c_{t,h'}))(\mathrm{Id} - \eta A^c(Z^c_{t,h'-1}))\cdots(\mathrm{Id} - \eta A^c(Z^c_{t,h}))$$

Analysis of FedLSA

We can characterize the bias of FedLSA:

$$\theta_t^{\mathrm{bias}} = \frac{1}{N}\sum_{c=1}^{N}(\mathrm{Id} - \bar{\Gamma}_{t,1:H})^{-1}(\mathrm{Id} - (\mathrm{Id} - \eta A^c)^H)\{\theta_\star^c - \theta_\star\}$$

where $\bar{\Gamma}_{t,1:H} = \frac{1}{N}\sum_{c=1}^{N}\Gamma_{t,1:H}^c$

Analysis of FedLSA

We can characterize the bias of FedLSA:

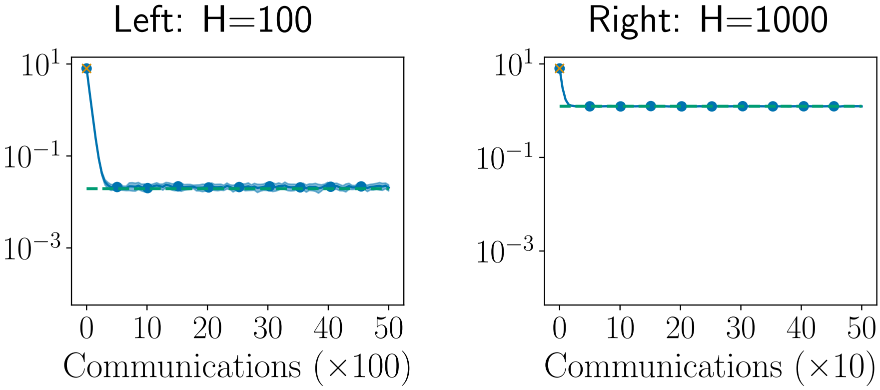
$$\theta_t^{\mathrm{bias}} = \frac{1}{N}\sum_{c=1}^{N}(\mathrm{Id} - \bar{\Gamma}_{t,1:H})^{-1}(\mathrm{Id} - (\mathrm{Id} - \eta A^c)^H)\{\theta_\star^c - \theta_\star\}$$

where $\bar{\Gamma}_{t,1:H} = \frac{1}{N}\sum_{c=1}^{N}\Gamma_{t,1:H}^c$

And give a convergence rate

$$\mathbb{E}\left[\|\theta_t - \theta_t^{bias} - \theta_\star\|^2\right] = O\left((1-\eta a)^{Ht}\|\theta_0 - \theta_\star\|^2 + \frac{\eta\sigma_\star^2}{Na}\right)$$

Numerical Illustration



Blue line: FedLSA's mean squared error
Green line: FedLSA's bias as predicted by our theory

Problem: heterogeneity requires lots of communications

To achieve $\mathbb{E}\left[\|\theta_T - \theta_\star\|^2\right] \leq \epsilon^2$, we need

- $\frac{\eta\sigma_\star^2}{Na} \leq \epsilon^2 \quad \rightarrow \eta = \frac{Na\epsilon^2}{\sigma_\star^2}$
- $\|\theta_T^{\text{bias}}\|^2 \leq \epsilon^2 \quad \rightarrow H = \frac{\sigma_\star^2}{N\epsilon\mathbb{E}_c[\|\theta_\star - \theta_\star^c\|]}$
- $(1-\eta a)^{Ht}\|\theta_0 - \theta_\star\|^2 \leq \epsilon^2 \quad \rightarrow T = \frac{\mathbb{E}_c[\|\theta_\star - \theta_\star^c\|]}{a^2\epsilon} \log \frac{\|\theta_0 - \theta_\star\|}{\epsilon}$

Solution: Control variates (SCAFFLSA)[5]

for $t = 0$ to $T - 1$ **do**

  Initialize $\theta_{t,0} = \theta_t$

  for each agent $c = 1..N$ **do**

    for $h = 1$ to H **do**

      Observe $Z^c_{t,h}$ and perform local update:

$$\theta_{t,h} = \theta^c_{t,h-1} - \eta(A^c(Z^c_{t,h})\theta^c_{t,h-1} - b^c(Z^c_{t,h}) - \xi_t)$$

    end for

  end for

  Aggregate local updates $\theta_{t+1} = \frac{1}{N}\sum_{c=1}^{N} \theta^c_{t,H}$

  Update control variate $\xi_{t+1} = \xi_t - \frac{1}{\eta H}(\theta_{t+1} - \theta^c_{t,H})$

end for

[5] Extending ideas from on Sai Praneeth Karimireddy et al. "Scaffold: Stochastic controlled averaging for federated learning". In: *ICML*. PMLR. 2020, pp. 5132–5143

Theoretical analysis

We prove, assuming $H \leq \frac{a}{\eta \max_c \|A^c\|^2}$

$$\mathbb{E}[\|\theta_T - \theta_\star\|^2] \lesssim \left(1 - \frac{\eta a H}{2}\right)^T \psi_0 + \frac{\eta \sigma_\star^2}{Na}$$

with $\psi_0 = \|\theta_0 - \theta_\star\|^2 + \eta^2 H^2 \mathbb{E}_c[\|A^c(\theta_\star^c - \theta_\star)\|^2]$

Theoretical analysis

We prove, assuming $H \leq \frac{a}{\eta \max_c \|A^c\|^2}$

$$\mathbb{E}[\|\theta_T - \theta_\star\|^2] \lesssim \left(1 - \frac{\eta a H}{2}\right)^T \psi_0 + \frac{\eta \sigma_\star^2}{Na}$$

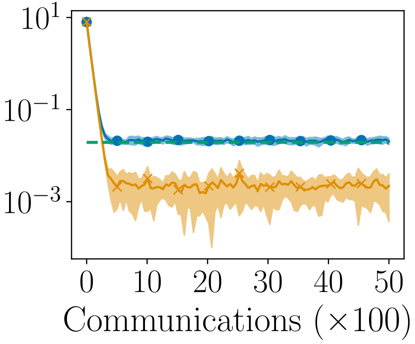
with $\psi_0 = \|\theta_0 - \theta_\star\|^2 + \eta^2 H^2 \mathbb{E}_c[\|A^c(\theta_\star^c - \theta_\star)\|^2]$

Note on analysis

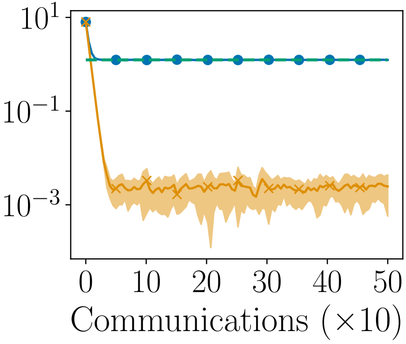
Direct analysis "à la LSA" does not work. We need a "Lyapunov" analysis, and to carefully study covariances of control variates to obtain linear speed-up.

Numerical Illustration

Left: H=100



Right: H=1000



Blue line: FedLSA's mean squared error
Orange line: SCAFFLSA's mean squared error

Communication Complexity

To achieve $\mathbb{E}\left[\|\theta_T - \theta_\star\|^2\right] \leq \epsilon^2$, we need

- $\frac{\eta\sigma_\star^2}{Na} \leq \epsilon^2$ $\quad\rightarrow \eta = \frac{Na\epsilon^2}{\sigma_\star^2}$
- $H \leq \frac{a}{\eta \max_c \|A^c\|^2}$ $\quad\rightarrow H = \frac{\sigma_\star^2}{N\epsilon^2 \max_c \|A^c\|^2}$
- $(1-\eta a)^{Ht}\|\theta_0 - \theta_\star\|^2 \leq \epsilon^2$ $\quad\rightarrow T = \frac{\max_c \|A^c\|^2}{a^2} \log \frac{\|\theta_0 - \theta_\star\|}{\epsilon}$

$\rightarrow H \propto 1/N\epsilon^2$ rather than $1/N\epsilon$, and T independent on ϵ

Parameter setting required to reach $\mathbb{E}\left[\|\theta_T - \theta_\star\|^2\right] \leq \epsilon^2$ for different algorithms/analyses

	Algorithm	Communication T	Local updates H	Sample complexity TH
	FedLSA[6]	$\mathcal{O}\left(\frac{N^2}{a^2\epsilon^2}\log\frac{1}{\epsilon}\right)$	1	$\mathcal{O}\left(\frac{N^2}{a^2\epsilon^2}\log\frac{1}{\epsilon}\right)$
new results	FedLSA	$\mathcal{O}\left(\frac{1}{a^2\epsilon}\log\frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{N\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{Na^2\epsilon^2}\log\frac{1}{\epsilon}\right)$
new results	Scaffnew [7]	$\mathcal{O}\left(\frac{1}{a\epsilon}\log\frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{a\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{a^2\epsilon^2}\log\frac{1}{\epsilon}\right)$
new results	Scafflsa	$\mathcal{O}\left(\frac{1}{a^2}\log\frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{N\epsilon^2}\right)$	$\mathcal{O}\left(\frac{1}{Na^2\epsilon^2}\log\frac{1}{\epsilon}\right)$

[6] Thinh T Doan. "Local stochastic approximation: A unified view of federated learning and distributed multi-task reinforcement learning algorithms". In: *arXiv:2006.13460* (2020).

[7] Adapted from Konstantin Mishchenko et al. "Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally!" In: *ICML*. 2022, pp. 15750–15769

Conclusion and Perspectives

Summary:

- We studied FedLSA's communication complexity
- We extended control variates methods to FedLSA
- We show that both methods have linear speed-up (up to bias)

Perspectives:

- SCAFFLSA's analysis is good in low step-size regimes: what about larger step sizes?
- Direct analysis of SCAFFLSA "à la FedLSA"?

Thank you!

Questions?

See the paper:

Paul Mangold et al. "SCAFFLSA: Taming Heterogeneity in Federated Linear Stochastic Approximation and TD Learning". In: *arXiv:2402.04114* (2024)