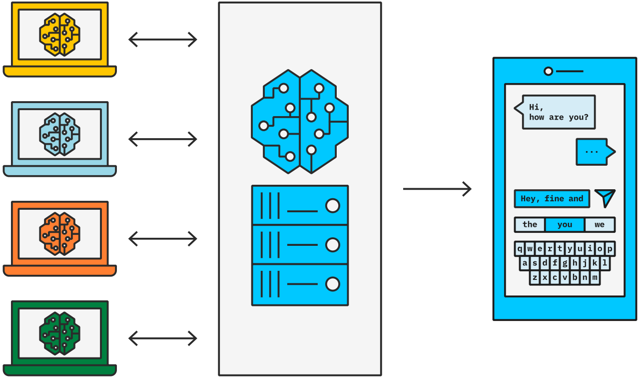
A Universal Framework for Federated (Convex) Optimization

Sebastian Stich | EUROPT | June 28, 2024

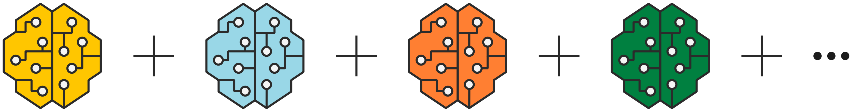
Federated Learning



- **private data stays on device**
- **server coordinates** training and aggregates focused updates

 [McM+17; Kai+21]

Federated Optimization Objective



$$\min_{\mathbf{x}\in\mathbb{R}^d}\left[f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})\right] \qquad \underbrace{f_i(\mathbf{x}) = \mathbb{E}_{\xi\sim\mathcal{D}_i} f_i(\mathbf{x},\xi)}_{\text{data } \mathcal{D}_i \text{ on client } i}$$

NB:

- If the client datasets are finite, we can also write $f_i(\mathbf{x}) = \frac{1}{|\mathcal{D}_i|}\sum_{\xi\in\mathcal{D}_i} f(\mathbf{x},\xi)$.
- **Cross-silo** setting when n is small; the **cross-device** setting when $n \to \infty$ is sometimes modeled as $f(\mathbf{x}) = \mathbb{E}_{i\sim\mathcal{C}} f_i(\mathbf{x})$.

Baseline I: Mini-Batch SGD

Init: $\mathbf{x}^0 \in \mathbb{R}^d$, stepsize $\gamma > 0$.

$$\mathbf{x}^{r+1} = \mathbf{x}^r - \frac{\gamma}{n} \sum_{i=1}^{n} \mathbf{g}_i(\mathbf{x}^r)$$

Where $\mathbf{g}_i$ denotes a stochastic gradient $\mathbb{E}_{\xi \sim \mathcal{D}_i} \mathbf{g}_i(\mathbf{x}) = \nabla f_i(\mathbf{x})$.

NB:

- In this talk we always assume $\nabla f_i(\mathbf{x})$ exists.
- Constraints can be added via a regularizer $\psi(\mathbf{x})$.
- The output is not necessarily the last iterate. Sometimes $\bar{\mathbf{x}} := \frac{1}{R+1} \sum_{r=0}^{R} \mathbf{x}^r$.

Mini-Batch SGD

Under the **assumptions**

- $f : \mathbb{R}^d \to \mathbb{R}$ is L-smooth, $\quad \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,
- $f : \mathbb{R}^d \to \mathbb{R}$ convex, $\quad f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.
- uniformly bounded variance, $\quad \mathbb{E} \|\mathbf{g}_i(\mathbf{x}) - \nabla f_i(\mathbf{x})\|^2 \leq \sigma^2, \forall \mathbf{x} \in \mathbb{R}^d$.

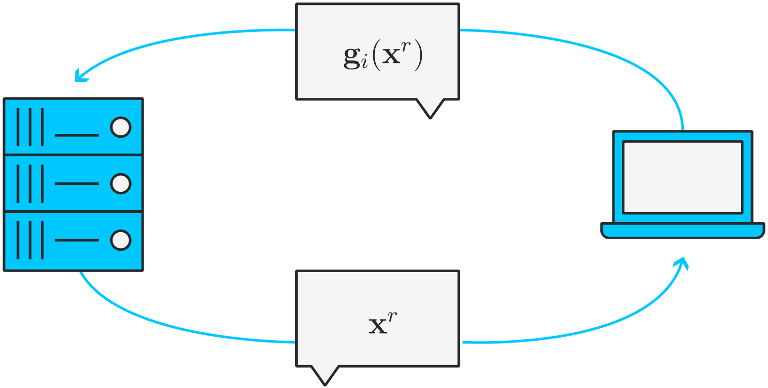
Mini-Batch SGD finds $\bar{\mathbf{x}}$ with $\mathbb{E} f(\bar{\mathbf{x}}) - f^\star \leq \epsilon$, where $f^\star = f(\mathbf{x}^\star), \mathbf{x}^\star \in \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$,

$$R = \mathcal{O}\left(\frac{\sigma^2}{n\epsilon^2} + \frac{L}{\epsilon} \left\|\mathbf{x}^0 - \mathbf{x}^\star\right\|^2 \right)$$

NB:

- Linear speedup in the number of devices n.

Communication Bottleneck



Performance measure:

communication complexity $\succ$ oracle calls

Baseline II: Acceleration

	communication rounds
Mini-Batch SGD	$\mathcal{O}\left(\frac{\sigma^2}{n\epsilon^2} + \frac{L}{\epsilon}\right)$
Accelerated Mini-Batch SGD	$\mathcal{O}\left(\frac{\sigma^2}{n\epsilon^2} + \sqrt{\frac{L}{\epsilon}}\right)$

It is not possible to converge in fewer iterations (standard lower bound for $n = 1$).

NB: In practice, simple gradient methods are often preferred over accelerated methods, due to robustness and **possibility to work on non-convex tasks!**

Is it possible to converge in fewer communication rounds?

Local Update Methods

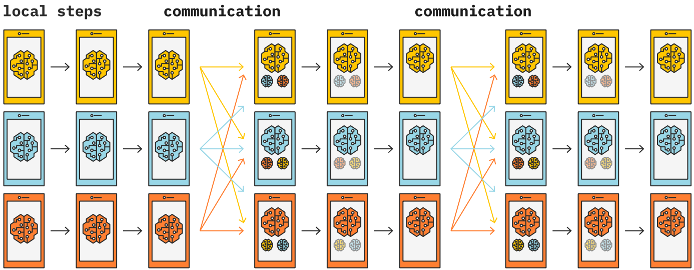
Local SGD

Init: $\mathbf{x}^0 \in \mathbb{R}^d$, stepsize $\gamma \geq 0$, local step parameter $K \geq 1$.

On each device, init $\mathbf{x}_{i,r,0} = \mathbf{x}^r$, and update K times:

$$\mathbf{x}_{i,r,k+1} = \mathbf{x}_{i,r,k} - \gamma \mathbf{g}_i(\mathbf{x}_{i,r,k})$$

On the server: $\mathbf{x}^{r+1} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_{i,r+1,K}$



 [McM+17; Sti19]

Local SGD - Advertisement

- Toy experiment:

ResNet-20 on CIFAR-10 (IID data)

	Top-1 acc.	local gradients	communication
Mini-batch SGD $(n = 16, \tau = 128)$	92.5%	2048	-
Mini-batch SGD $(n = 16, \tau = 1024)$	76.3%	16384	$\div$ 8
Local-SGD $(n = 16, \tau = 8 \times 128)$	92.0%	16384	$\div$ 8

- Federated Averaging ($\approx$ Local SGD) is used for decentralized machine learning across industries.
  - healthcare data analysis
  - smart devices
  - finance
  - . . .

 [Lin+20]

FedProx [Li+20]

Init: $\mathbf{x}^0 \in \mathbb{R}^d$, $\lambda > 0$.

On each device:

$$\mathbf{x}_{i,r+1} \approx \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^d} \left\{ F_{i,r}(\mathbf{x}) := f_i(\mathbf{x}) + \underbrace{\frac{\lambda}{2}\left\|\mathbf{x} - \mathbf{x}^r\right\|^2}_{\text{proximal point}} \right\}$$

On the sever: $\mathbf{x}^{r+1} := \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_{i,r+1}$.

- Intuitively (not rigorously) $\lambda \approx \frac{1}{K\gamma}$.
- **Can be combined with arbitrary local solvers!**

• Local SGD ($\mathbb{E}\mathbf{g}_i = \nabla f_i$)	• Gradient Descent (∇f_i)	• Accelerated Gradient Methods

 [Ber11; Li+20]

Some History

Optimization difficulty: drift

Observation: $\mathbf{x}^\star$ is not a fixed point for $K \geq 2$!

- **Intuitive explanation**:
  Local SGD steps overfit on the local data: $\mathbf{x}_{i,r,k} \overset{k\to\infty}{\longrightarrow} \left[\mathbf{x}_i^\star := \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^d} f_i(\mathbf{x})\right]$.

$$\mathbf{x}^\star \neq \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i^\star$$

- **Mathematical explanation**:
  Concretely, consider $f_1(x) = \frac{1}{2}x^2$, $f_2(x) = (x-1)^2$ with $\mathbf{x}^0 = \mathbf{x}^\star = \frac{2}{3}$. After one round with $K = 2$ steps it holds

$$\mathbf{x}^1 = \frac{1}{2}\left(\mathbf{x}_{1,0,K} + \mathbf{x}_{2,0,K}\right) = \frac{1}{2}\left(\frac{2}{3}(1-\gamma)^2 + 1 - \frac{1}{3}(1-2\gamma)^2\right) = \frac{2}{3} - \frac{\gamma^2}{3} \neq \frac{2}{3} = \mathbf{x}^\star$$

First-Order Similarity

The functions $f_1, \ldots, f_n$ are $\zeta_\star^2$-*similar* if

$$\frac{1}{n}\sum_{i=1}^{n} \left\|\nabla f_i(\mathbf{x}^\star) - \nabla f(\mathbf{x}^\star)\right\|^2 \leq \zeta_\star^2$$

NB:

- For quadratic functions $f_i(\mathbf{x}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{A}_i\mathbf{x}\rangle - \langle \mathbf{b}_i, \mathbf{x}\rangle + \mathbf{c}_i$, $f(\mathbf{x}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle - \langle \mathbf{b}, \mathbf{x}\rangle + \mathbf{c}$.

$$\zeta_\star^2 := \frac{1}{n}\sum_{i=1}^{n}\left\|\nabla f(\mathbf{x}^\star)\right\|^2 = \frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{A}_i\mathbf{x}^\star - \mathbf{b}_i\right\|^2 = \frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{A}_i\mathbf{A}^\dagger\mathbf{b} - \mathbf{b}_i\right\|^2$$

- For the special case when $\mathbf{A}_i \equiv \mathbf{A}$, then $\zeta_\star^2 = \frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{b} - \mathbf{b}_i\right\|^2$.

 [Kol+20]

Local SGD Convergence

[Kol+20]. There exists a $\gamma > 0$ such that $\mathbb{E} f(\mathbf{x}^R) - f^\star \leq \epsilon$ for

$$R = \mathcal{O}\left(\frac{\sigma^2}{nK\epsilon^2} + \frac{\sqrt{L}\left(\zeta_\star + \sigma/\sqrt{K}\right)}{\epsilon^{3/2}} + \frac{L}{\epsilon}\left\|\mathbf{x}_0 - \mathbf{x}^\star\right\|^2 \right)$$

NB:

- Asymptotically, linear speedup in nK.
- Can be slower than mini-batch SGD.
- Does not reflect well the practical performance:
- [Pat+24]: result is optimal under these assumptions.

On the Unreasonable Effectiveness of Federated Averaging with Heterogeneous Data

Jianyu Wang Carnegie Mellon University
Rudrajit Das University of Texas at Austin
Gauri Joshi Carnegie Mellon University
Satyen Kale Google Research
Zheng Xu Google Research
Tong Zhang Google Research and HKUST

 [Kol+20; Wan+22; Pat+24]

Drift-Correction

A series of works developed **drift-correction** mechanisms of the form

$$\mathbf{x}_{i,r,k+1} = \mathbf{x}_{i,r,k} - \gamma \left(\underbrace{\nabla f_i(\mathbf{x})}_{\text{normal update}} + \underbrace{\mathbf{c}_r - \mathbf{c}_{r,i}}_{\text{drift correction}} \right)$$

where $\mathbf{c}_r, \mathbf{c}_{r,i} \in \mathbb{R}^d$ are suitably (learned) corrections.

Observe that for $\mathbf{c}_r = \nabla f(\mathbf{x}^\star)$, $\mathbf{c}_{r,i} = \nabla f_i(\mathbf{x}^\star)$, the optimal solution becomes again a fixed point:

$$\mathbf{x}^\star = \mathbf{x}^\star - (\nabla f_i(\mathbf{x}^\star) + \nabla f(\mathbf{x}^\star) - \nabla f_i(\mathbf{x}^\star))$$

The communication complexity of methods like:

- SCAFFOLD [Kar+20]
- ProxSkip [Mis+22]

does not (or only weakly) depend on $\zeta_\star$.

Stabilized Proximal Point Method

[ ] **Xiaowen Jiang, Anton Rodomanov** and **S.**
Stabilized Proximal Point Methods for Federated Optimization, 2024.

Second-order similarity

Measure how the *relatedness* $h_i(\mathbf{x}) := f_i(\mathbf{x}) - f(\mathbf{x})$ changes:

The functions $f_1, \ldots, f_n$ are δ-*similar* if

$$\frac{1}{n}\sum_{i=1}^{n} \left\|\nabla h_i(\mathbf{x}) - \nabla h_i(\mathbf{y})\right\|^2 \leq \delta^2 \left\|\mathbf{x} - \mathbf{y}\right\|^2 \qquad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$$

NB:

- For quadratic functions $\|\nabla h_i(\mathbf{x}) - \nabla h_i(\mathbf{y})\| = \|(\mathbf{A}_i - \mathbf{A})(\mathbf{x} - \mathbf{y})\|$. $f_i(\mathbf{x}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{A}_i \mathbf{x}\rangle - \langle \mathbf{b}_i, \mathbf{x}\rangle + \mathbf{c}_i$, $f(\mathbf{x}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle - \langle \mathbf{b}, \mathbf{x}\rangle + \mathbf{c}$.
- δ-similarity holds if $\frac{1}{n}\sum_{i=1}^{n} \left\|\nabla^2 h_i(\mathbf{x})\right\|^2 \leq \delta^2$, $\forall \mathbf{x} \in \mathbb{R}^d$.
- If each f_i is L-smooth, then $\delta \leq L$.

 [AS15; Kar+21; Kov+22; KJ23]

Can we prove that the communication rounds decrease for small δ?

DANE [SSZ14]

Init: $\mathbf{x}^0 \in \mathbb{R}^d$, $\lambda > 0$.

On each device:

$$\mathbf{x}_{i,r+1} \approx \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^d} \left\{ F_{i,r}(\mathbf{x}) := f_i(\mathbf{x}) + \underbrace{\langle \mathbf{x} - \mathbf{x}^r, \nabla f(\mathbf{x}^r) - \nabla f_i(\mathbf{x}^r)\rangle}_{\text{drift correction}} + \underbrace{\frac{\lambda}{2} \|\mathbf{x} - \mathbf{x}^r\|^2}_{\text{proximal point}} \right\}$$

On the sever: $\mathbf{x}^{r+1} := \frac{1}{n}\sum_{i=1}^n \mathbf{x}_{i,r+1}$.

[JRS24a]. Let $\lambda = \Theta(\delta)$, and assume each f_i is L-smooth. Then $f(\bar{\mathbf{x}}^R) - f^\star \leq \epsilon$ for

$$R = \mathcal{O}\left(\frac{\delta}{\epsilon}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2\right)$$

 [SSZ14; JRS24a]

DANE

	communication rounds
Fast Gradient Method	$\mathcal{O}\left(\sqrt{\frac{L}{\epsilon}}\right)$
DANE	$\mathcal{O}\left(\frac{\delta}{\epsilon}\right)$
FedProx	$\mathcal{O}\left(\frac{\sqrt{L}\zeta_\star}{\epsilon^{3/2}} + \frac{L}{\epsilon}\right)$

- DANE be can faster than FGM when $\delta \leq \sqrt{L\epsilon}$.
- Mitigates drift (no dependence on $\zeta_\star$).
- **Allows to use arbitrary local solvers!**

• Local SGD ($\mathbb{E}\mathbf{g}_i = \nabla f_i$)	• Gradient Descent (∇f_i)	• Accelerated Gradient Methods

Caveat: the local subproblem

The local probimal subproblem in round r needs to be solved with accuracy:

$$\|\nabla F_{i,r}(\mathbf{x}_{i,r+1})\| \leq \Theta\left(\frac{\lambda}{r}\right)\|\mathbf{x}_{i,r+1} - \mathbf{x}^r\|$$

Local Gradient Oracle Calls in round r:

$$\mathcal{O}\left(\sqrt{\frac{L}{\delta}\ln r}\right)$$

Stabilized Proximal Point (= S-DANE) [new]

Init:

On each device:

$$\mathbf{x}_{i,r+1} \approx \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^d} \left\{ F_{i,r}(\mathbf{x}) := f_i(\mathbf{x}) + \underbrace{\langle \mathbf{x} - \mathbf{x}^r, \nabla f(\mathbf{v}^r) - \nabla f_i(\mathbf{v}^r)\rangle}_{\text{drift correction}} + \underbrace{\frac{\lambda}{2}\|\mathbf{x} - \mathbf{v}^r\|^2}_{\text{proximal point}} \right\}$$

On the sever: $\mathbf{x}^{r+1} := \frac{1}{n}\sum_{i=1}^n \mathbf{x}_{i,r+1}$,

$$\underbrace{\mathbf{v}^{r+1} = \mathbf{v}^r - \frac{1}{\lambda n}\sum_{i=1}^n \nabla f_i(\mathbf{x}_{i,r+1})}_{\text{extra-gradient step}}$$

NB: For $n = 1$ (and without drift correction) related to

- an instance of the hybrid projection-proximal point algorithm [SS99] (non-accelerated version of [MS13])
- Extra-gradient [Nes23]

Stabilized Proximal Point

[JRS24b]. Let $\lambda = \Theta(\delta)$, and assume each f_i is L-smooth. Then $f(\bar{\mathbf{x}}^R) - f^\star \leq \epsilon$ for

$$R = \mathcal{O}\left(\frac{\delta}{\epsilon}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2\right) \qquad \text{and} \qquad K = \mathcal{O}\left(\sqrt{\frac{L}{\delta}}\right)$$

NB:

- the inner complexity is e.g. reached for the optimized gradient method [KF18].
- For the gradient method $K = \mathcal{O}\left(\frac{L}{\delta}\right)$.

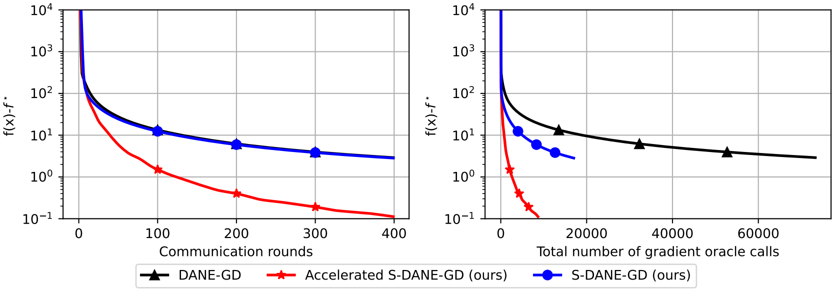
Stabilized Proximal Point: Discussion

	communication rounds	inner iterations
Fast Gradient Method	$\mathcal{O}\left(\sqrt{\frac{L}{\epsilon}}\right)$	$\mathcal{O}(1)$
Stabilized DANE	$\mathcal{O}\left(\frac{\delta}{\epsilon}\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\delta}}\right)$
+ acceleration	$\mathcal{O}\left(\sqrt{\frac{\delta}{\epsilon}}\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\delta}}\right)$

NB: [Kov+22] derive a similar result with a gradient-sliding method, under a different definition of δ', with $\delta' \geq \delta$.

Numerical Illustration

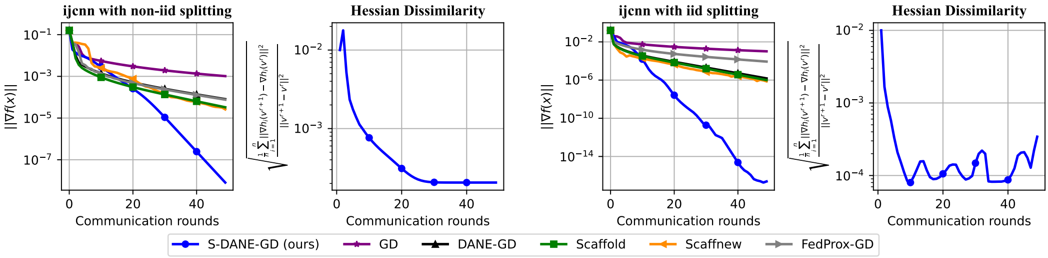
Speedup in communication rounds and gradient oracle calls:



$f_i(\mathbf{x}) := \frac{1}{m}\sum_{j=1}^{m}\frac{1}{2}(\mathbf{x}-\mathbf{b}_{i,j})^T\mathbf{A}_{i,j}(\mathbf{x}-\mathbf{b}_{i,j})$ where $\mathbf{b}_{i,j} \in \mathbb{R}^d$ and $\mathbf{A}_{i,j} \in \mathbb{R}^{d\times d}$. We use $n = 10$, $m = 5$, $d = 1000$ and generate $\max_{i,j}\{\|A_{i,j}\|\} = 100$ and $\delta \approx 5$. All three methods use GD as the local solver.

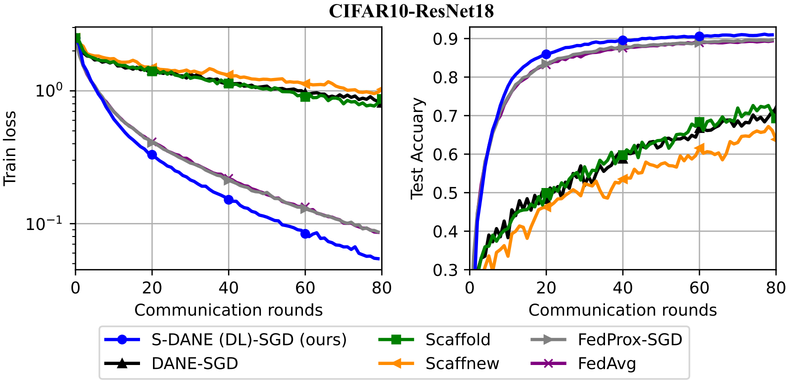
Illustration: Adaptive stepsize λ

Adaptive stepsize $\lambda_r = \sqrt{\frac{\frac{1}{n}\sum_{i=1}^{n}\|\nabla h_i(\mathbf{v}^r) - \nabla h_i(\mathbf{v}^{r-1})\|^2}{\|\mathbf{v}^r - \mathbf{v}^{r-1}\|^2}}.$



logistic regression: $f_i(\mathbf{x}) := \frac{n}{M}\sum_{j=1}^{m_i}\log(1+\exp(-y_{i,j}\mathbf{a}_{i,j}^T\mathbf{x})) + \frac{1}{2M}\|\mathbf{x}\|^2$ on the **ijcnn** dataset

Illustration: Distributed Neural Network Training



Discussion

Discussion

- **General framework**: clients can use arbitrary local solvers.
- **"Stabilized Catalyst"**: For $n = 1$ acceleration framework similar to Catalyst Acceleration, without the $\mathcal{O}\left(\ln\left(\frac{1}{\epsilon}\right)\right)$ overhead.
- **Client Sampling** is also possible.

References

Main references

[JRS24a] X. Jiang, A. Rodomanov, and S. U. Stich. **Federated Optimization with Doubly Regularized Drift Correction.** In: International conference on machine learning. 2024.

[JRS24b] X. Jiang, A. Rodomanov, and S. U. Stich. **Stabilized Proximal Point Methods for Federated Optimization.** In: Advances in Neural Information Processing Systems (2024).

[Kar+20] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh. **Scaffold: Stochastic controlled averaging for federated learning.** In: International conference on machine learning. PMLR. 2020, pp. 5132–5143.

[Kol+20] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich. **A unified theory of decentralized sgd with changing topology and local updates.** In: International Conference on Machine Learning. PMLR. 2020, pp. 5381–5393.

[Sti19] S. U. Stich. **Local SGD Converges Fast and Communicates Little.** In: International Conference on Learning Representations. 2019.

References

[AS15] Y. Arjevani and O. Shamir. **Communication complexity of distributed convex learning and optimization.** In: Advances in neural information processing systems 28 (2015).

[Ber11] D. P. Bertsekas. **Incremental proximal methods for large scale convex optimization.** In: Mathematical programming 129.2 (2011), pp. 163–195.

[Kai+21] 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.** In: Foundations and trends® in machine learning 14.1–2 (2021), pp. 1–210.

[Kar+21] S. P. Karimireddy, M. Jaggi, S. Kale, M. Mohri, S. Reddi, S. U. Stich, and A. T. Suresh. **Breaking the centralized barrier for cross-device federated learning.** In: Advances in Neural Information Processing Systems 34 (2021), pp. 28663–28676.

[KF18] D. Kim and J. A. Fessler. **Generalizing the optimized gradient method for smooth convex minimization.** In: SIAM Journal on Optimization 28.2 (2018), pp. 1920–1950.

[KJ23] A. Khaled and C. Jin. **Faster federated optimization under second-order similarity.** In: International Conference on Learning Representations. 2023.

[Kov+22] D. Kovalev, A. Beznosikov, E. Borodich, A. Gasnikov, and G. Scutari. **Optimal gradient sliding and its application to optimal distributed optimization under similarity.** In: Advances in Neural Information Processing Systems 35 (2022), pp. 33494–33507.

[Li+20] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. **Federated optimization in heterogeneous networks.** In: Proceedings of Machine learning and systems 2 (2020), pp. 429–450.

[Lin+20] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi. **Don't Use Large Mini-batches, Use Local SGD.** In: International Conference on Learning Representations. 2020.

[McM+17] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. **Communication-efficient learning of deep networks from decentralized data.** In: Artificial intelligence and statistics. PMLR. 2017, pp. 1273–1282.

[Mis+22] K. Mishchenko, G. Malinovsky, S. Stich, and P. Richtárik. **Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally!** In: International Conference on Machine Learning. PMLR. 2022, pp. 15750–15769.

[MS13] R. D. C. Monteiro and B. F. Svaiter. **An Accelerated Hybrid Proximal Extragradient Method for Convex Optimization and Its Implications to Second-Order Methods.** In: SIAM Journal on Optimization 23.2 (2013), pp. 1092–1125.

[Nes23] Y. Nesterov. **High-Order Reduced-Gradient Methods for Composite Variational Inequalities.** In: arXiv preprint arXiv:2311.15154 (2023).

[Pat+24] K. K. Patel, M. Glasgow, A. Zindari, L. Wang, S. U. Stich, Z. Cheng, N. Joshi, and N. Srebro. **The limits and potentials of local sgd for distributed heterogeneous learning with intermittent communication.** In: arXiv preprint arXiv:2405.11667 (2024).

[SS99] M. Solodov and B. Svaiter. **A hybrid projection-proximal point algorithm.** eng. In: Journal of Convex Analysis 6.1 (1999), pp. 59–70.

[SSZ14] O. Shamir, N. Srebro, and T. Zhang. **Communication-efficient distributed optimization using an approximate newton-type method.** In: International conference on machine learning. PMLR. 2014, pp. 1000–1008.

[Wan+22] J. Wang, R. Das, G. Joshi, S. Kale, Z. Xu, and T. Zhang. **On the unreasonable effectiveness of federated averaging with heterogeneous data.** In: arXiv preprint arXiv:2206.04723 (2022).