Optimization for Machine Learning in Practice II

Martin Jaggi

EPFL

Machine Learning and Optimization Laboratory

mlo.epfl.ch

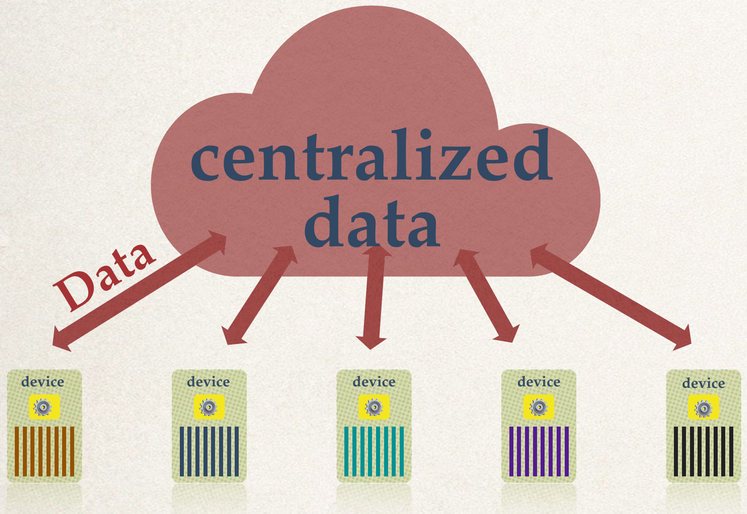
2

Collaborative Learning

Evolution

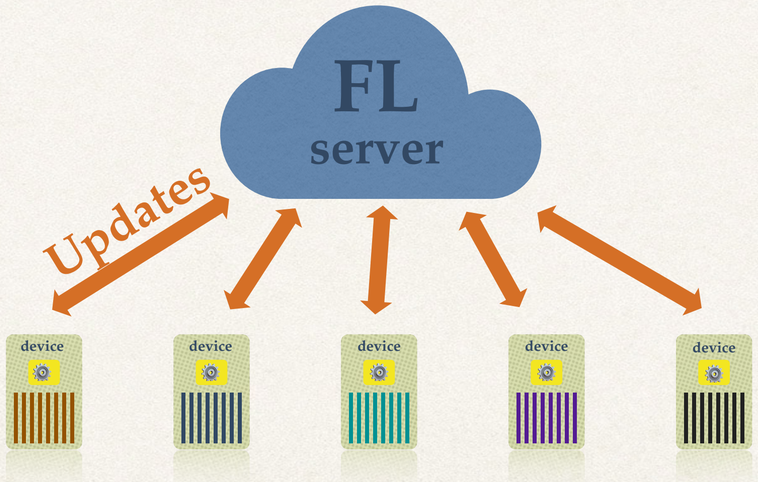
centralized

traditional, sharing data



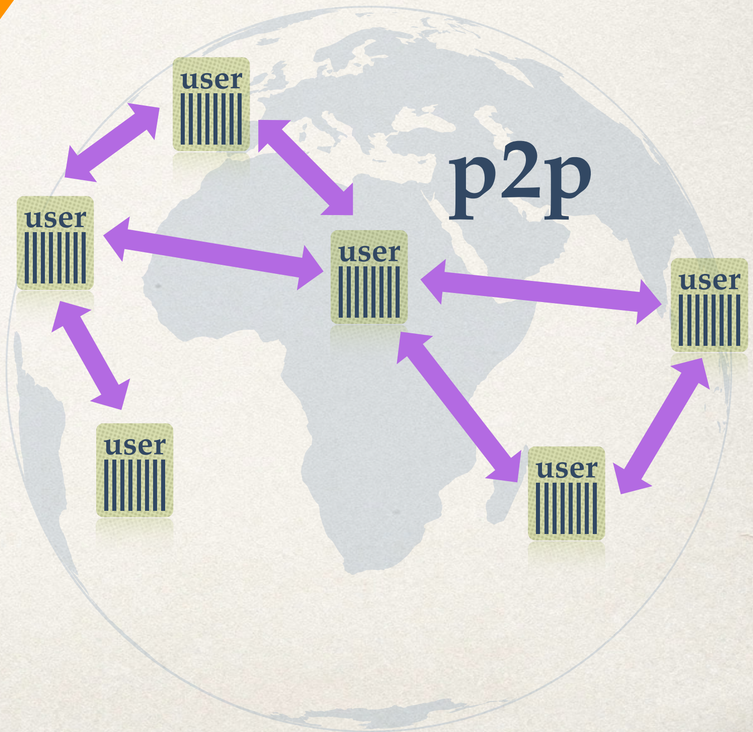
federated

sharing model updates



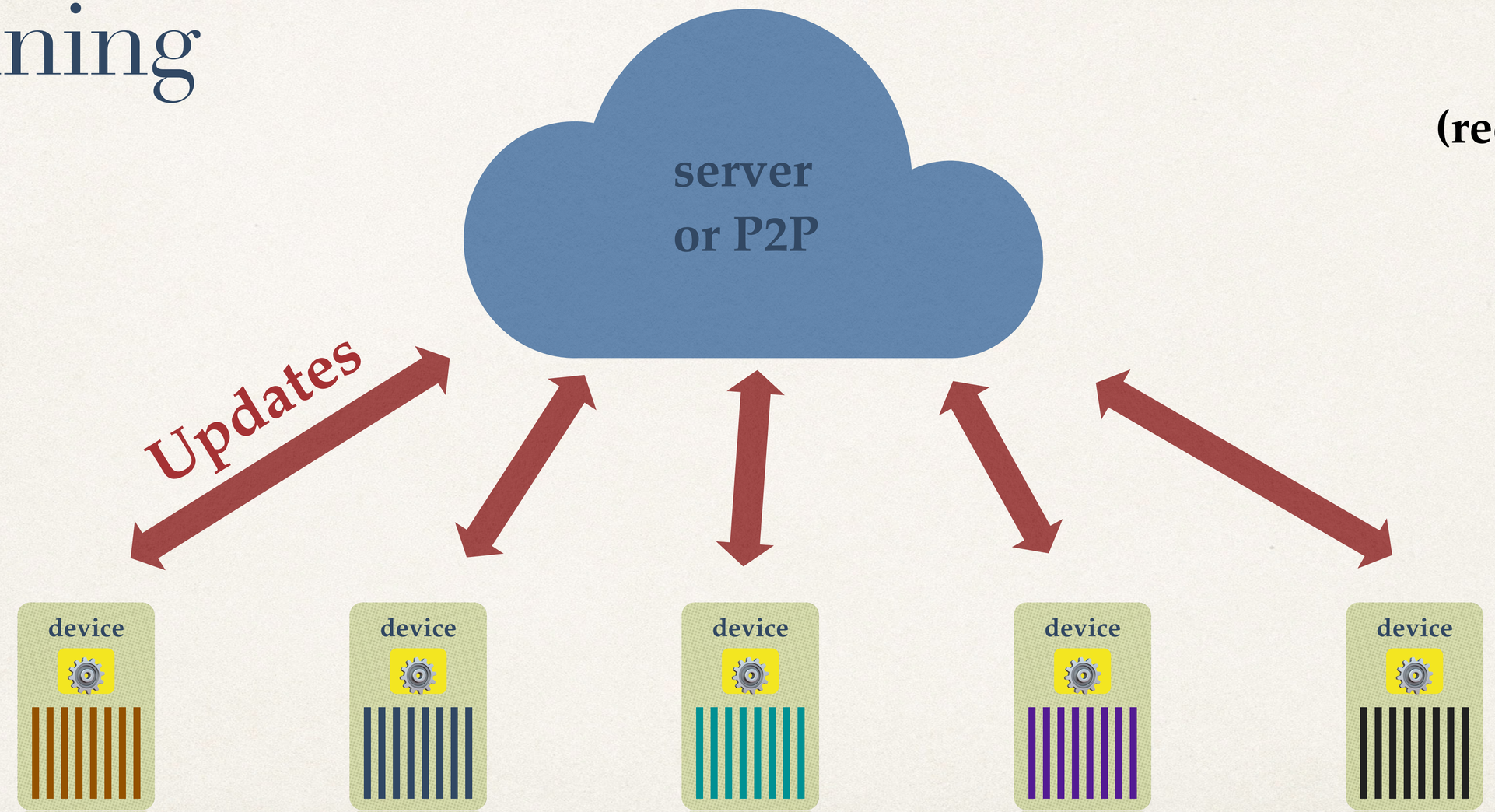
decentralized

collaborative learning

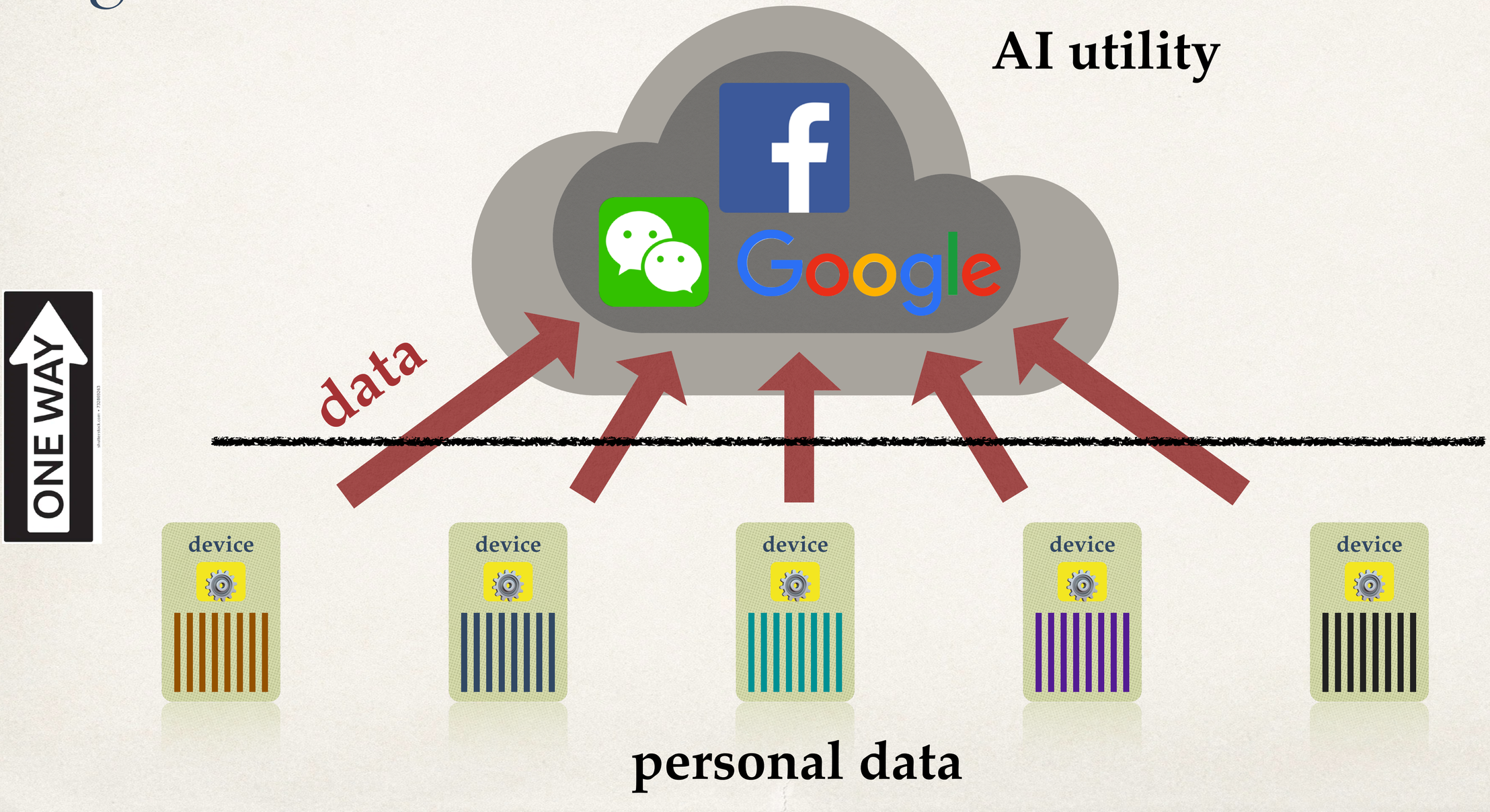


Collaborative & Federated Training

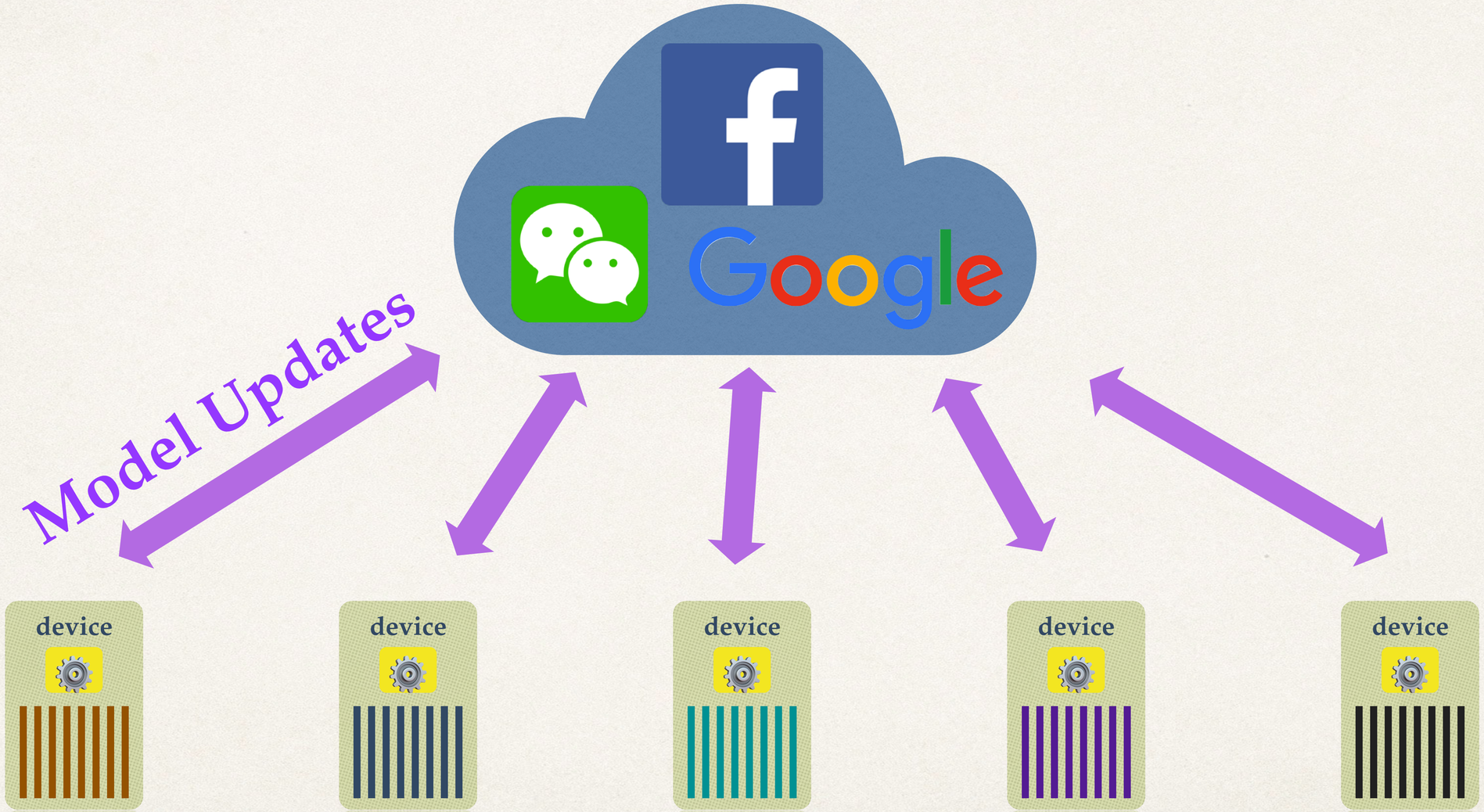
(recap)



Big Picture

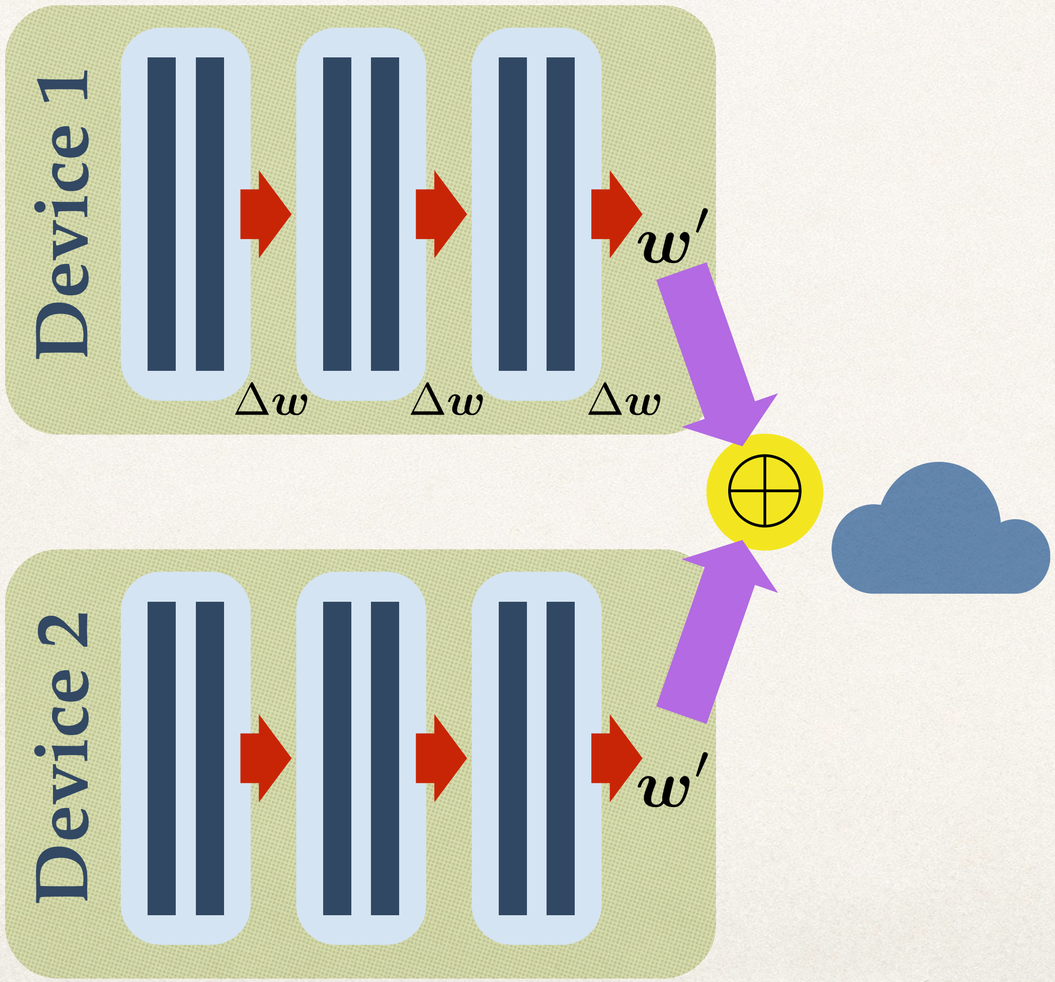

AI utility
Google
data
ONE WAY
device
device
device
device
device
personal data

2a Federated Learning



data never leaves device

2a Federated Learning



- Local SGD steps = "Federated averaging"
- Google Android Keyboard

Training Objective

Slides by Sebastian Stich

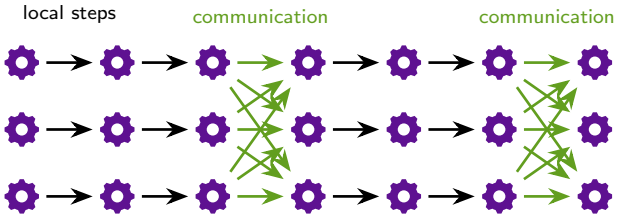
$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} \underbrace{f_i(\mathbf{x})}_{\text{data } \mathcal{D}_i \text{ on client } i} \right] \qquad f_i(\mathbf{x}) = \begin{cases} \mathbb{E}_{\xi\sim\mathcal{D}_i} F(\mathbf{x},\xi) \\ \frac{1}{m}\sum_{j=1}^{m} f_{ij}(\mathbf{x})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

- Collaboratively solve **a (joint)** machine learning problem
- **efficiently**, in terms of:
  - computation (stochastic gradients, mini-batches),
  - communication (server $\leftrightarrow$ client).

Other very relevant scenarios:

• personalization • heterogeneity • privacy • robustness

Local SGD



Notation/Setting

- n machines
- $f_i(\mathbf{x})$ denote the function (data) available locally at node i
- local gradient oracle $\mathbb{E}[\mathbf{g}^{(i)}(\mathbf{x})] = \nabla f_i(\mathbf{x})$, $\forall i \in [n]$, with bounded variance:

$$\mathbb{E} \left\| \mathbf{g}^{(i)}(\mathbf{x}) - \nabla f_i(\mathbf{x}) \right\|^2 \leq \sigma^2$$

- let $\mathbf{x}_t^{(i)} \in \mathbb{R}^d$ denote the local iterate at node i, $\forall i \in [n]$

Local SGD

Input: $\mathbf{x}_0 \in \mathbb{R}^d$, $\mathbf{x}_0^{(i)} = \mathbf{x}_0$, $\forall i \in [n]$, stepsize γ, $\tau \geq 1$ (number of local steps)

At iteration t (in parallel on all nodes $i \in [n]$):

$\mathbf{g}_t^i = \mathbf{g}^{(i)}\left(\mathbf{x}_t^{(i)}\right)$ (stochastic gradient locally on each node)

if $t+1$ is a multiple of τ :

$$\mathbf{x}_{t+1}^{(i)} = \frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_t^{(i)} - \gamma \mathbf{g}_t^i\right)$$ (global averaging)

otherwise:

$$\mathbf{x}_{t+1}^{(i)} = \mathbf{x}_t^{(i)} - \gamma \mathbf{g}_t^i$$ (local step)

Homogeneous Data

For simplicity, assume

$$f_1(\mathbf{x}) = f_2(\mathbf{x}) = \cdots = f_n(\mathbf{x})$$

(note that in general $\mathbf{g}_t^i \neq \mathbf{g}_t^j$ for $i \neq j$).

- This means stochastic gradients are uniformly sampled from the whole dataset (similar as for mini-batch SGD).
- For $\tau = 1$ Local SGD is identical to mini-batch SGD with batch size $b = n$.

Theorem (Homogeneous Case, [Sti18, KLB+20])

Let $f_i \colon \mathbb{R}^d \to \mathbb{R}$ be L-smooth, $\forall i \in [n]$ and $f_i = f_j$, $\forall i, j \in [n]$, with $\Delta = f(\mathbf{x}_0) - f^\star$. Then there exists a stepsize $\gamma \leq \gamma_{\text{crit}} := \frac{1}{20L\tau}$ such that after T steps (that is, T/τ communication rounds) of Local SGD it holds

$$\min_{t \leq T} \mathbb{E} \left\| \nabla f(\bar{\mathbf{x}}_t) \right\|^2 = \mathcal{O}\left(\frac{\Delta L \tau}{T} + \frac{(\Delta L \sigma)^{2/3} \tau^{1/3}}{T^{2/3}} + \frac{\sqrt{L \Delta \sigma^2}}{\sqrt{Tn}} \right),$$

with $\bar{\mathbf{x}}_t := \frac{1}{n} \sum_{i=1}^n \mathbf{x}_t^{(i)}$.

Discussion

- Linear speedup if $\sigma^2 > 0$: the variance decreases linearly in the number of oracle calls (Tn). This is optimal.
- The deterministic optimization term (the term not depending on σ^2) is impacted by τ (similarly as with mini-batch SGD).
  - ideally, we would have hoped to see there $\mathcal{O}\left(\frac{\Delta L}{T}\right)$ (= progress in every iteration) vs. $\mathcal{O}\left(\frac{\Delta L \tau}{T}\right)$ (= progress every communication round)
- The theorem shows almost the same convergence as for mini-batch SGD, up to the higher order $\mathcal{O}(T^{-2/3})$ term.
  - There is no clear winner.

Performance in Practice [LSPJ19]

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

Proof I

We will use the virtual sequence technique. As virtual sequence we consider $\bar{\mathbf{x}}_t$ (note that the average is not computed in every iteration).

Lemma (Decrease)

For $\gamma \leq \frac{1}{4L}$ it holds

$$\mathbb{E} f(\bar{\mathbf{x}}_{t+1}) \leq E f(\bar{\mathbf{x}}_t) - \frac{\gamma}{4} \left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \gamma^2 L \frac{\sigma^2}{n} + \frac{\gamma L^2}{n} \sum_{i=1}^{n} \mathbb{E} \left\|\mathbf{x}_t^{(i)} - \bar{\mathbf{x}}_t\right\|^2$$

Lemma (Difference)

For $\gamma \leq \gamma_{\text{crit}} = \frac{1}{20L\tau}$, with the notation for $R_t = \frac{1}{n} \sum_{i=1}^{n} \left\|\bar{\mathbf{x}}_t - \mathbf{x}_t^{(i)}\right\|^2$, it holds

$$\mathbb{E} R_t \leq \frac{1}{20L^2\tau} \sum_{j=(t-1)-k}^{t-1} \mathbb{E} \left\|\nabla f(\bar{\mathbf{x}}_j)\right\|^2 + 5\gamma^2 \tau \sigma^2$$

where $(t-1)-k$ denotes the index of the last communication round ($k \leq \tau - 1$).

Proof II

Plug (Difference) into (Decrease), re-arrange and divide by γ:

$$\frac{1}{4}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 \leq \frac{1}{\gamma}\left(\mathbb{E}f(\bar{\mathbf{x}}_t) - \mathbb{E}f(\bar{\mathbf{x}}_{t+1})\right) + \gamma L\frac{\sigma^2}{n} + \frac{1}{20\tau}\sum_{j=(t-1)-k}^{t-1}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_j)\right\|^2 + 5\gamma^2L^2\tau\sigma^2$$

Now we divide by T and sum over $t = 0, \ldots, T-1$:

$$\frac{1}{4T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 \leq \frac{1}{\gamma T}\sum_{t=0}^{T-1}\left[\left(\mathbb{E}f(\bar{\mathbf{x}}_t) - \mathbb{E}f(\bar{\mathbf{x}}_{t+1})\right) + \frac{1}{20T}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2\right] + \gamma^2L\frac{\sigma^2}{n} + 5\gamma L^2\tau\sigma^2$$

Note that $\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2$ appears on both sides, with $\frac{1}{4T} - \frac{1}{20T} = \frac{1}{5T}$.

$$\frac{1}{5T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 \leq \frac{\Delta}{\gamma T} + \gamma L\frac{\sigma^2}{n} + 5\gamma^2L^2\tau\sigma^2\,.$$

Now the result follows by tuning γ (Exercise).

Proof of Lemma (Decrease)

By L-smoothness:

$$\begin{aligned}\mathbb{E}f(\bar{\mathbf{x}}_{t+1}) &\leq \mathbb{E}f(\bar{\mathbf{x}}_t) - \tfrac{\gamma}{n}\textstyle\sum_{i=1}^n \nabla f(\bar{\mathbf{x}}_t)^\top \mathbf{g}_t^i + \tfrac{\gamma^2 L}{2}\mathbb{E}\left\|\tfrac{1}{n}\textstyle\sum_{i=1}^n \mathbf{g}_t^i\right\|^2 \\ &\leq \mathbb{E}f(\bar{\mathbf{x}}_t) - \tfrac{\gamma}{n}\textstyle\sum_{i=1}^n \nabla f(\bar{\mathbf{x}}_t)^\top \nabla f_i(\mathbf{x}_t^{(i)}) + \tfrac{\gamma^2 L\sigma^2}{2n} + \tfrac{\gamma^2 L}{2}\mathbb{E}\left\|\tfrac{1}{n}\textstyle\sum_{i=1}^n \nabla f_i(\mathbf{x}_t^{(i)})\right\|^2\end{aligned}$$

Similarly as we have seen before, by adding and subtracting $\nabla f(\bar{\mathbf{x}}_t)$:

$$-\tfrac{1}{n}\textstyle\sum_{i=1}^n \nabla f(\bar{\mathbf{x}}_t)^\top \nabla f_i(\mathbf{x}_t^{(i)}) = -\nabla f(\bar{\mathbf{x}}_t)^\top \nabla f(\bar{\mathbf{x}}_t) + \tfrac{1}{n}\textstyle\sum_{i=1}^n \nabla f(\bar{\mathbf{x}}_t)^\top \left(\nabla f(\bar{\mathbf{x}}_t) - \nabla f_i(\mathbf{x}_t^{(i)})\right)$$

$$\leq -\tfrac{1}{2}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \tfrac{1}{2}\left\|\nabla f(\bar{\mathbf{x}}_t) - \tfrac{1}{n}\textstyle\sum_{i=1}^n \nabla f_i(\mathbf{x}_t^{(i)})\right\|^2$$

$$\leq -\tfrac{1}{2}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \tfrac{L^2}{2n}\textstyle\sum_{i=1}^n \left\|\bar{\mathbf{x}}_t - \mathbf{x}_t^{(i)}\right\|^2$$

Note: $-\mathbf{a}^\top\mathbf{b} \leq \frac{1}{2}\|\mathbf{a}\|^2 + \frac{1}{2}\|\mathbf{b}\|^2$.

Continued

And by adding and subtracting $\nabla f(\bar{\mathbf{x}}_t)$ in the last term: $\|\mathbf{a}+\mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$

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

Now we plug everything together, and use $\gamma \leq \frac{1}{4L}$.

$$\mathbb{E}f(\bar{\mathbf{x}}_{t+1}) \leq \mathbb{E}f(\bar{\mathbf{x}}_t) + \left(\gamma^2 L - \frac{\gamma}{2}\right)\|\nabla f(\bar{\mathbf{x}}_t)\|^2 + \frac{\gamma^2 L\sigma^2}{2n} + \left(\frac{\gamma L^2}{2} + \gamma^2 L^3\right)\frac{1}{n}\sum_{i=1}^{n}\left\|\bar{\mathbf{x}}_t - \mathbf{x}_t^{(i)}\right\|^2$$

Proof of Lemma (Difference)

Note that if t is a multiple of τ, then $R_t = 0$ and there is nothing to prove. Otherwise note that

$$\mathbb{E}R_{t+1} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\bar{\mathbf{x}}_{t+1} - \mathbf{x}_{t+1}^i\right\|^2 = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\bar{\mathbf{x}}_t - \mathbf{x}_t^i + \gamma\mathbf{g}_t^i - \gamma\bar{\mathbf{g}}_t\right\|^2$$
$$\leq \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\bar{\mathbf{x}}_t - \mathbf{x}_t^i + \gamma\nabla f_i(\mathbf{x}_t^i) - \gamma\bar{\mathbf{v}}_t\right\|^2 + \gamma^2\sigma^2\,,$$

where $\bar{\mathbf{g}}_t := \frac{1}{n}\sum_{i=1}^n \mathbf{g}_t^i$ and $\bar{\mathbf{v}}_t := \frac{1}{n}\sum_{i=1}^n \nabla f_i(\mathbf{x}_t^{(i)})$ denote the average of the client gradients. With the inequality $\|\mathbf{a}+\mathbf{b}\|^2 \leq (1+\tau^{-1})\|\mathbf{a}\|^2 + 2\tau\|\mathbf{b}\|^2$ for $\tau \geq 1$, $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$, we continue:

$$\mathbb{E}R_{t+1} \leq \left(1+\frac{1}{\tau}\right)\mathbb{E}R_t + \frac{2\tau\gamma^2}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\nabla f(\mathbf{x}_t^{(i)}) - \bar{\mathbf{v}}_t\right\|^2 + \gamma^2\sigma^2$$
$$\leq \left(1+\frac{1}{\tau}\right)\mathbb{E}R_t + \frac{2\tau\gamma^2}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\nabla f_i(\mathbf{x}_t^{(i)})\right\|^2 + \gamma^2\sigma^2$$
$$\leq \left(1+\frac{1}{\tau}\right)\mathbb{E}R_t + \frac{2\tau\gamma^2}{n}\sum_{i=1}^{n}\mathbb{E}\left(2\left\|\nabla f_i(\bar{\mathbf{x}}_t)\right\|^2 + 2\left\|\nabla f_i(\mathbf{x}_t^{(i)}) - \nabla f_i(\bar{\mathbf{x}}_t)\right\|^2\right) + \gamma^2\sigma^2$$

$(*)$

Continued

We now use $f_1 = f_2 = \cdots = f_n$ and $\gamma \leq \frac{1}{20L\tau}$ to simplify:

$$\begin{aligned}\mathbb{E}R_{t+1} &\leq \left(1 + \frac{1}{\tau}\right)\mathbb{E}R_t + \frac{1}{100L^2\tau}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \frac{1}{100\tau}\mathbb{E}R_t + \gamma^2\sigma^2 \\ &\leq \left(1 + \frac{3}{2\tau}\right)\mathbb{E}R_t + \frac{1}{100L^2\tau}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \gamma^2\sigma^2\end{aligned}$$

The lemma now follows by unrolling, and noting that $\left(1 + \frac{3}{2\tau}\right)^j \leq 5$ for all $0 \leq j \leq \tau$.

Federated Learning on Heterogeneous Data

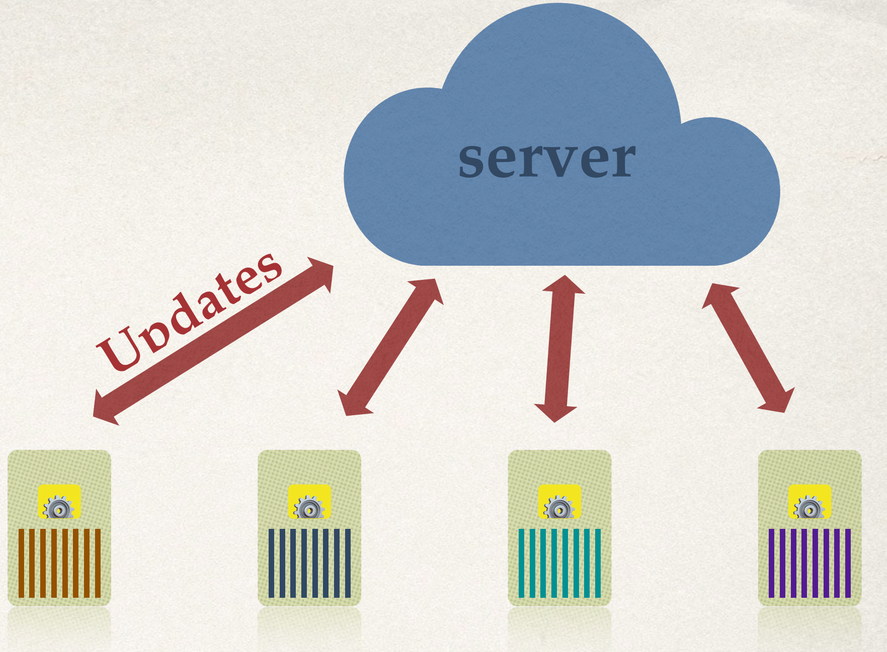
- We now want/need to drop the assumption that $f_i(\mathbf{x}) = f_j(\mathbf{x})$, $i \neq j$.

Data-heterogeneity causes drift when doing local steps.

Client drift

- **Federated Learning**

$$\min_{x} \frac{1}{n} \sum_{i}^{n} f_i(x)$$

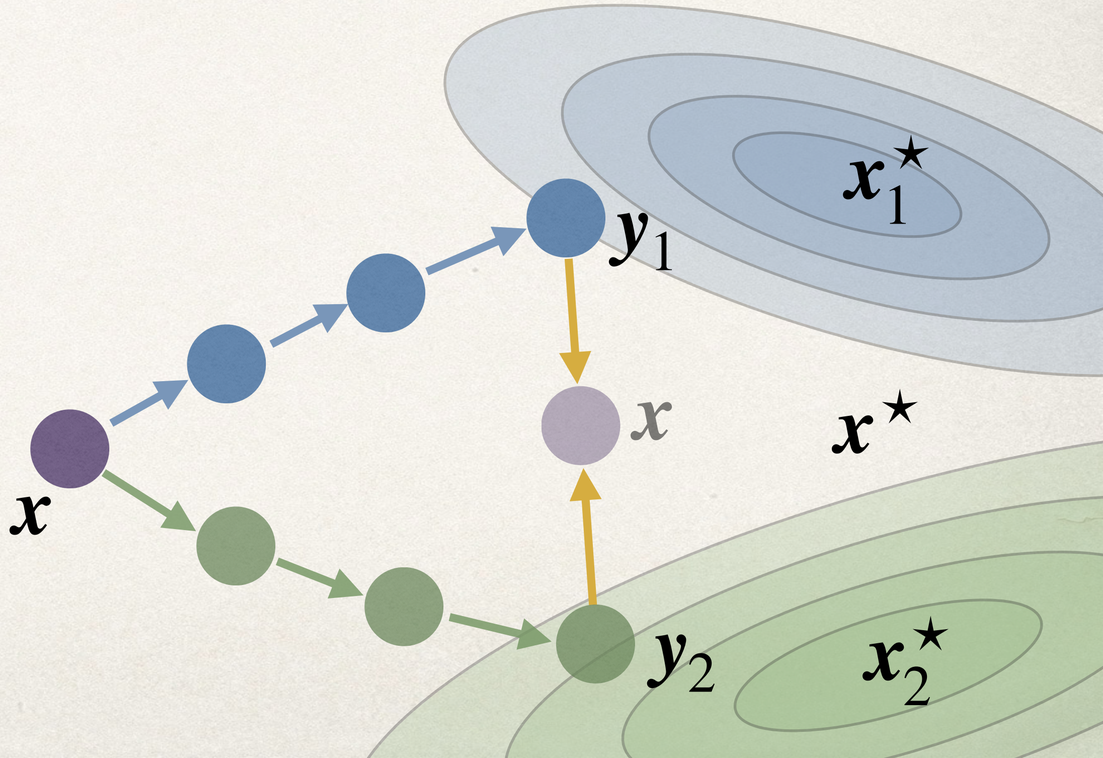


- **Fed Avg / Local SGD**

for some local steps

$$y_i := y_i - \eta \nabla f_i(y_i)$$

$$x := \frac{1}{n} \sum_{i=1}^{n} y_i \qquad \textit{(aggregation)}$$



Measuring Data Heterogeneity

Definition (Heterogeneity ζ^2)

(The smallest) parameter ζ^2 such that for all $\mathbf{x} \in \mathbb{R}^d$:

$$\frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \leq \zeta^2 \,.$$

Similar to the definition of the variance, but now we measure the inter-worker variance.

Convergence

Theorem (Heterogeneous Case, [KLB+20])

Let $f_i \colon \mathbb{R}^d \to \mathbb{R}$ be L-smooth, $\forall i \in [n]$ and the function heterogeneity bounded by ζ^2, with $\Delta = f(\mathbf{x}_0) - f^\star$. Then for there exists a stepsize $\gamma \leq \gamma_{\text{crit}} := \frac{1}{10L\tau}$ such that after T steps (that is, T/τ communication rounds) of Local SGD it holds

$$\min_{t \leq T} \mathbb{E} \left\| \nabla f(\bar{\mathbf{x}}_t) \right\|^2 = \mathcal{O}\left(\frac{\Delta L \tau}{T} + \left(\frac{L\Delta(\tau\zeta + \sqrt{\tau}\sigma}{T} \right)^{2/3} + \frac{\sqrt{L\Delta\sigma^2}}{\sqrt{Tn}} \right),$$

with $\bar{\mathbf{x}}_t := \frac{1}{n} \sum_{i=1}^n \mathbf{x}_t^{(i)}$.

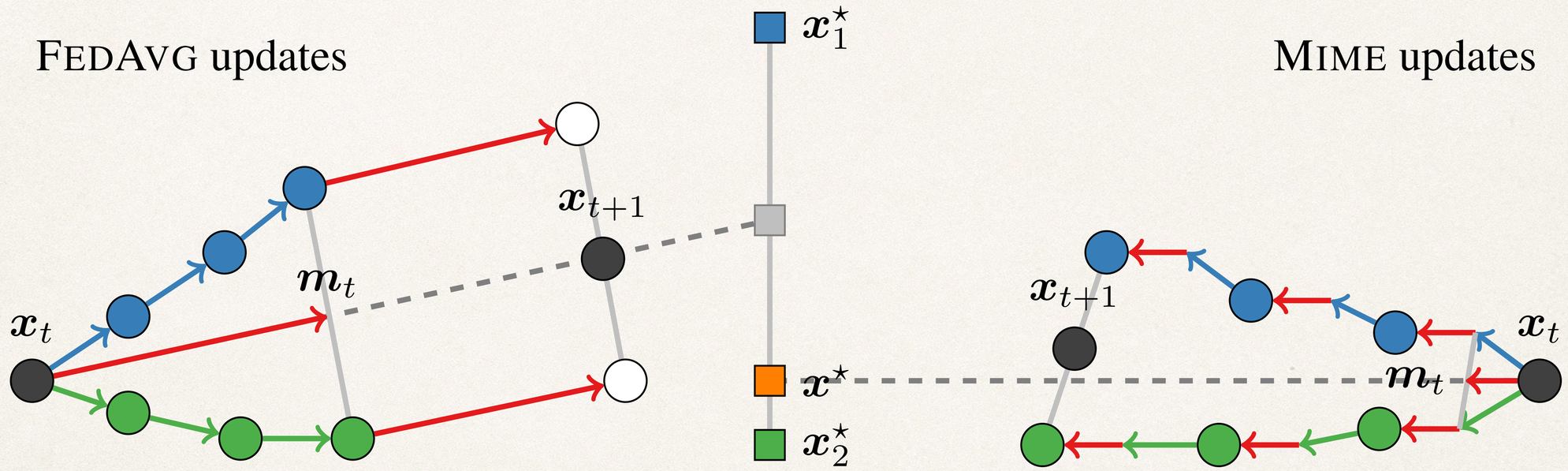
Discussion

- $\zeta^2 > 0$ slows down the convergence.
- The dependency on ζ is optimal.
- In general, the convergence is slower than for mini-batch SGD.
- Momentum (applied on the server) speeds up convergence, and solves the drift problem [KJK+21, CHY23]

Why does then Local SGD behave well on many practical problems?

- Speedup can be proven under different similarity assumptions.

Client drift



Mime algorithm framework

for *some local steps*

$$\boldsymbol{y}_i := \boldsymbol{y}_i - \eta\Big((1-\beta)\,\nabla f_i(\boldsymbol{y}_i) + \beta\boldsymbol{m}\Big)$$

$$\boldsymbol{m} := (1-\beta)\,\nabla f_i(\boldsymbol{x}) + \beta\boldsymbol{m}$$

aggregated on server after each round

Proof

We will prove a new (Difference) Lemma, the rest of the proof will follow the usual pattern.

Lemma (Difference)

For $\gamma \leq \gamma_{\text{crit}} = \frac{1}{20L\tau}$, *with the notation for* $R_t = \frac{1}{n}\sum_{i=1}^{n} \left\| \bar{\mathbf{x}}_t - \mathbf{x}_t^{(i)} \right\|^2$, *it holds*

$$\mathbb{E}R_t \leq \frac{1}{10L^2\tau} \sum_{j=(t-1)-k}^{t-1} \mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_j)\right\|^2 + 5\gamma^2\sigma^2\tau + 40\gamma^2\tau^2\zeta^2$$

where $(t-1)-k$ *denotes the index of the last communication round* ($k \leq \tau - 1$).

Proof II

Plug the new (Difference) bound into (Decrease), re-arrange, divide by γ, divide by T, and sum over $t = 0, \dots, T-1$... (left as exercise!)

We will end up with:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 = \mathcal{O}\left(\frac{\Delta}{\gamma T} + \gamma L\frac{\sigma^2}{n} + \gamma^2 L^2(\tau^2\zeta^2 + \tau\sigma^2)\right)$$

Now use previous exercise.

Proof of Lemma (Difference) I

For the proof of the difference lemma, we need to be more careful at inequality $(*)$. Recall the inequality $(*)$:

$$\mathbb{E}R_{t+1} \leq \left(1 + \frac{1}{\tau}\right) \mathbb{E}R_t + \frac{2\tau\gamma^2}{n} \sum_{i=1}^{n} \mathbb{E}\left(2\left\|\nabla f_i(\bar{\mathbf{x}}_t)\right\|^2 + 2\left\|\nabla f_i(\mathbf{x}_t^{(i)}) - \nabla f_i(\bar{\mathbf{x}}_t)\right\|^2\right) + \gamma^2\sigma^2$$

Previously we used $f_i = f_j$, to simplify. Instead, note that:

$$\begin{aligned}
\tfrac{1}{n}\textstyle\sum_{i=1}^{n}\left\|\nabla f_i(\bar{\mathbf{x}}_j)\right\|^2 &= \tfrac{1}{n}\textstyle\sum_{i=1}^{n}\left(\left\|\nabla f_i(\bar{\mathbf{x}}_j) - \nabla f(\bar{\mathbf{x}}_j) + \nabla f(\bar{\mathbf{x}}_j)\right\|^2\right) \\
&\leq \tfrac{2}{n}\textstyle\sum_{i=1}^{n}\left(\left\|\nabla f_i(\bar{\mathbf{x}}_j) - \nabla f(\bar{\mathbf{x}}_j)\right\|^2 + \left\|\nabla f(\bar{\mathbf{x}}_j)\right\|^2\right) \\
&\leq 2\zeta^2 + 2\left\|\nabla f(\bar{\mathbf{x}}_j)\right\|^2
\end{aligned}$$

Proof of Lemma (Difference) II

Therefore (and with $\gamma \leq \frac{1}{20L\tau}$)

$$\begin{aligned}\mathbb{E}R_{t+1} &\leq \left(1+\frac{1}{\tau}\right)\mathbb{E}R_t + \frac{1}{50L^2\tau}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + \frac{1}{100\tau}\mathbb{E}R_t + 8\tau\gamma^2\zeta^2 + \gamma^2\sigma^2 \\ &\leq \left(1+\frac{3}{2\tau}\right)\mathbb{E}R_t + \frac{1}{50L^2\tau}\mathbb{E}\left\|\nabla f(\bar{\mathbf{x}}_t)\right\|^2 + 8\tau\gamma^2\zeta^2 + \gamma^2\sigma^2\end{aligned}$$

The lemma now follows by unrolling, and noting that $\left(1+\frac{3}{2\tau}\right)^j \leq 5$ for all $0 \leq j \leq \tau$.

Bibliography I

Ziheng Cheng, Xinmeng Huang, and Kun Yuan.
Momentum benefits non-iid federated learning simply and provably.
arXiv preprint arXiv:2306.16504, 2023.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U Stich, and Ananda Theertha Suresh.
Mime: Mimicking centralized stochastic algorithms in federated learning.
In *NeurIPS*, 2021.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian U. Stich.
A unified theory of decentralized SGD with changing topology and local updates.
In *ICML 2020 - International Conference on Machine Learning*, pages 5381–5393, 2020.

Tao Lin, Sebastian U Stich, Kumar Kshitij Patel, and Martin Jaggi.
Don't Use Large Mini-batches, Use Local SGD.
In *ICLR - International Conference on Learning Representations*, 2019.

Bibliography II


Sebastian U Stich.
Local SGD Converges Fast and Communicates Little.
In *ICLR - International Conference on Learning Representations*, 2018.

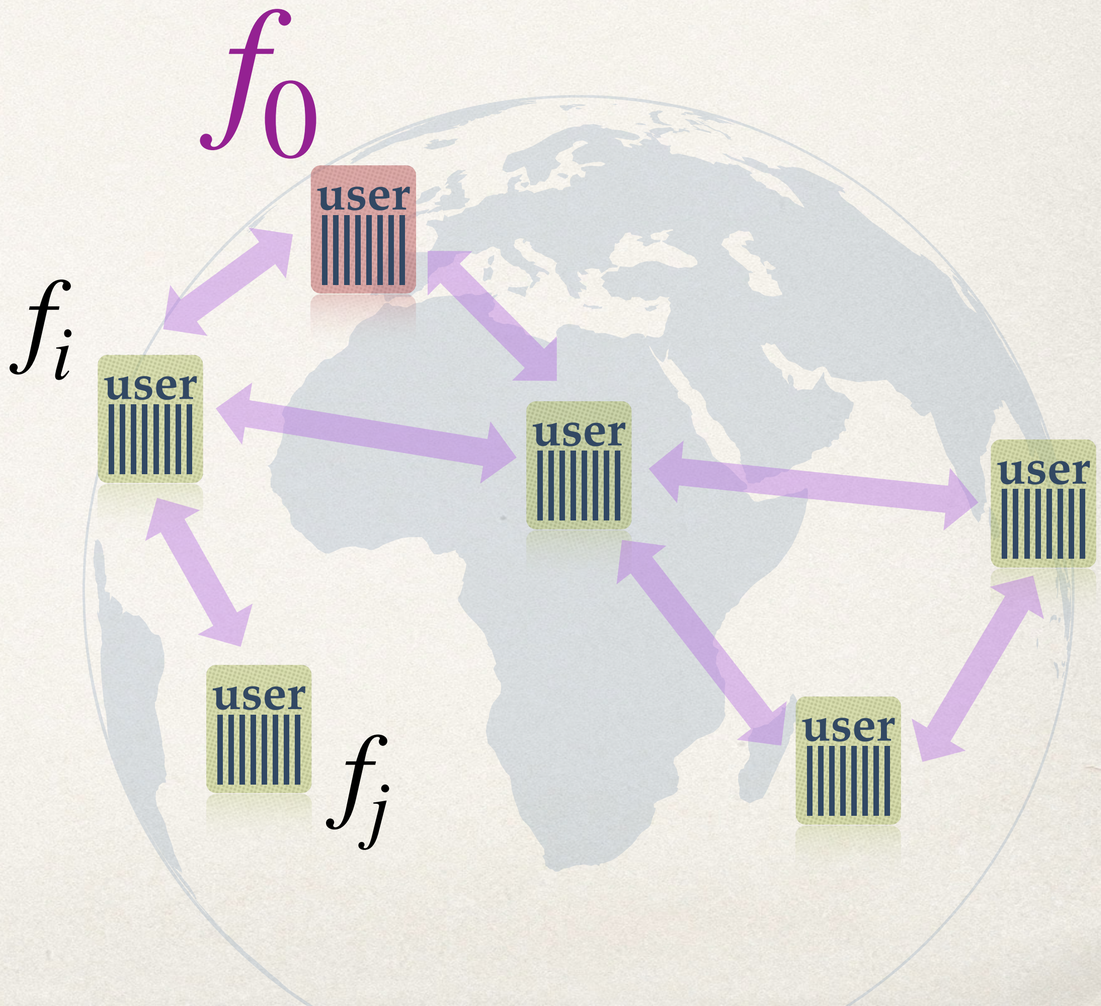
2b Federated vs Personalized Learning

- **Federated**

$$\min_{x} \frac{1}{n} \sum_{i}^{n} f_i(x)$$

- **Collaborative / Personalized**

$$\min_{x} f_0(x) \quad \Big| \quad \min_{x} f_1(x), \quad \min_{x} f_n(x)$$



2b Federated vs Personalized Learning

- **Federated**

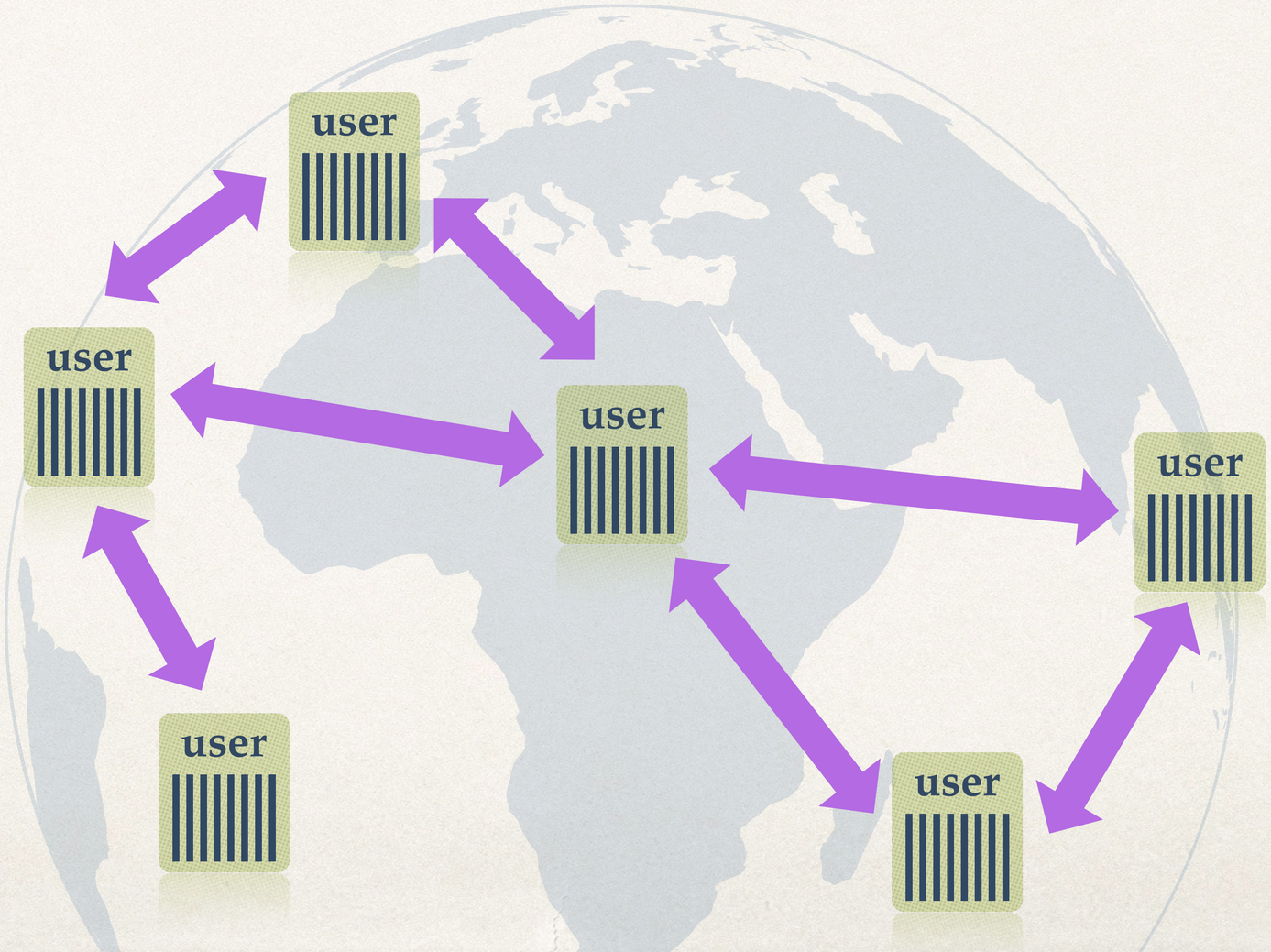
$$\min_{x} \frac{1}{n} \sum_{i}^{n} f_i(x)$$

- **Collaborative / Personalized**

$$\min_{x} f_0(x) \quad \Big| \quad \min_{x} f_1(x), \quad \ldots, \quad \min_{x} f_n(x)$$

- **Ordering of training**
  Set of active clients evolves (how?)
- **Clients = Tasks**
  Sequential fine-tuning
  Transfer learning,
  overparameterized models?
- **Train alone or collaborate?**

2c Decentralized Learning



Motivation

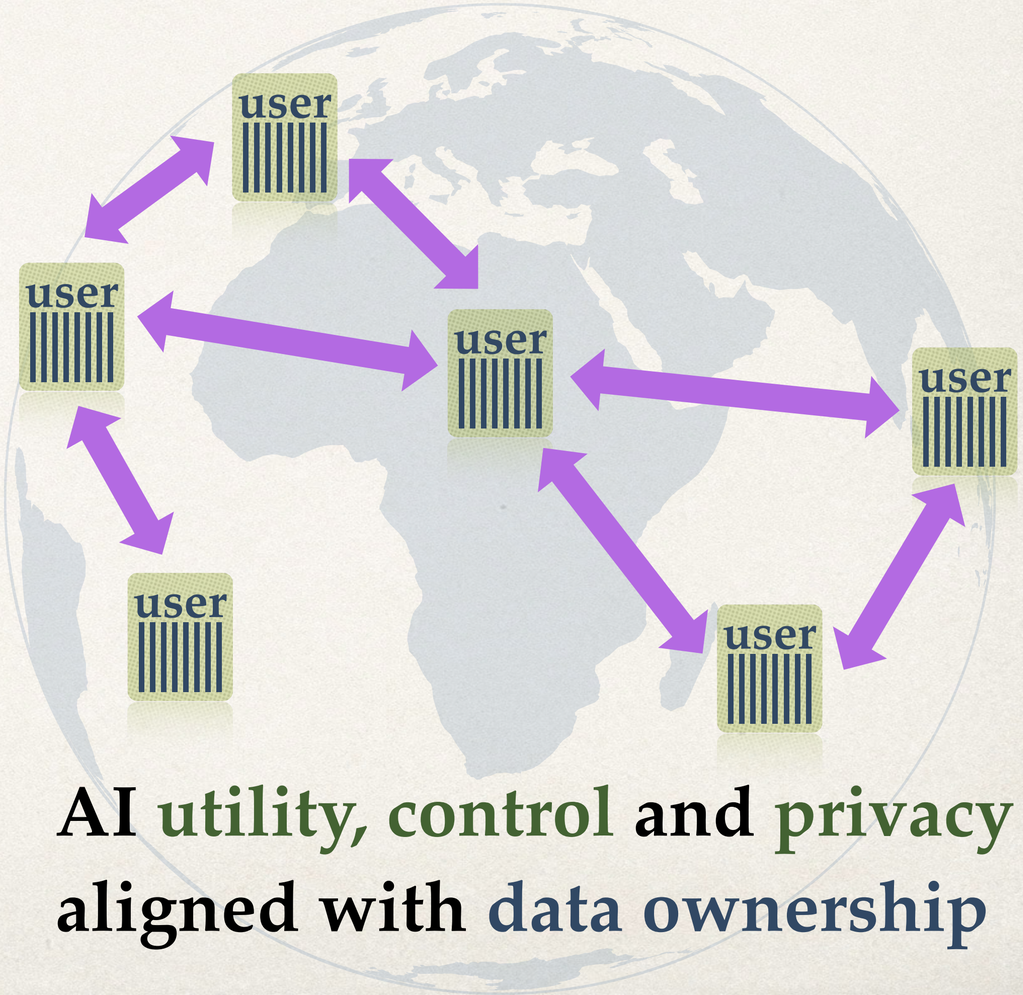
- **Applications:**

  any ML system with user data

  servers, devices, sensors, hospitals, ...

image source

- **Advantages:**



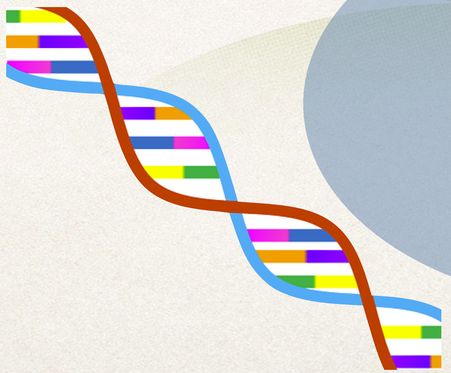
AI utility, control and privacy aligned with data ownership

Required Building Blocks

Robustness

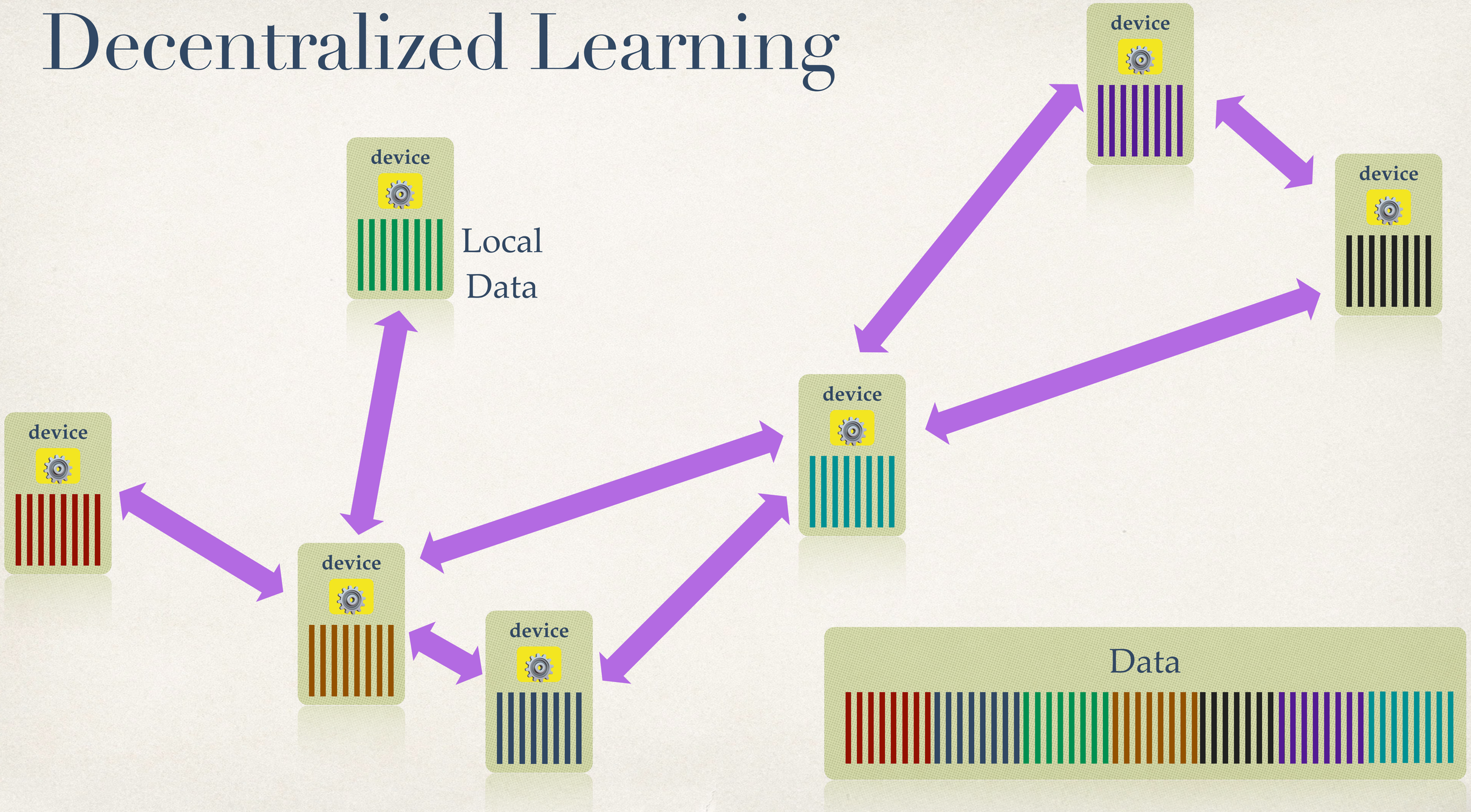
Efficiency

Decentralized ML

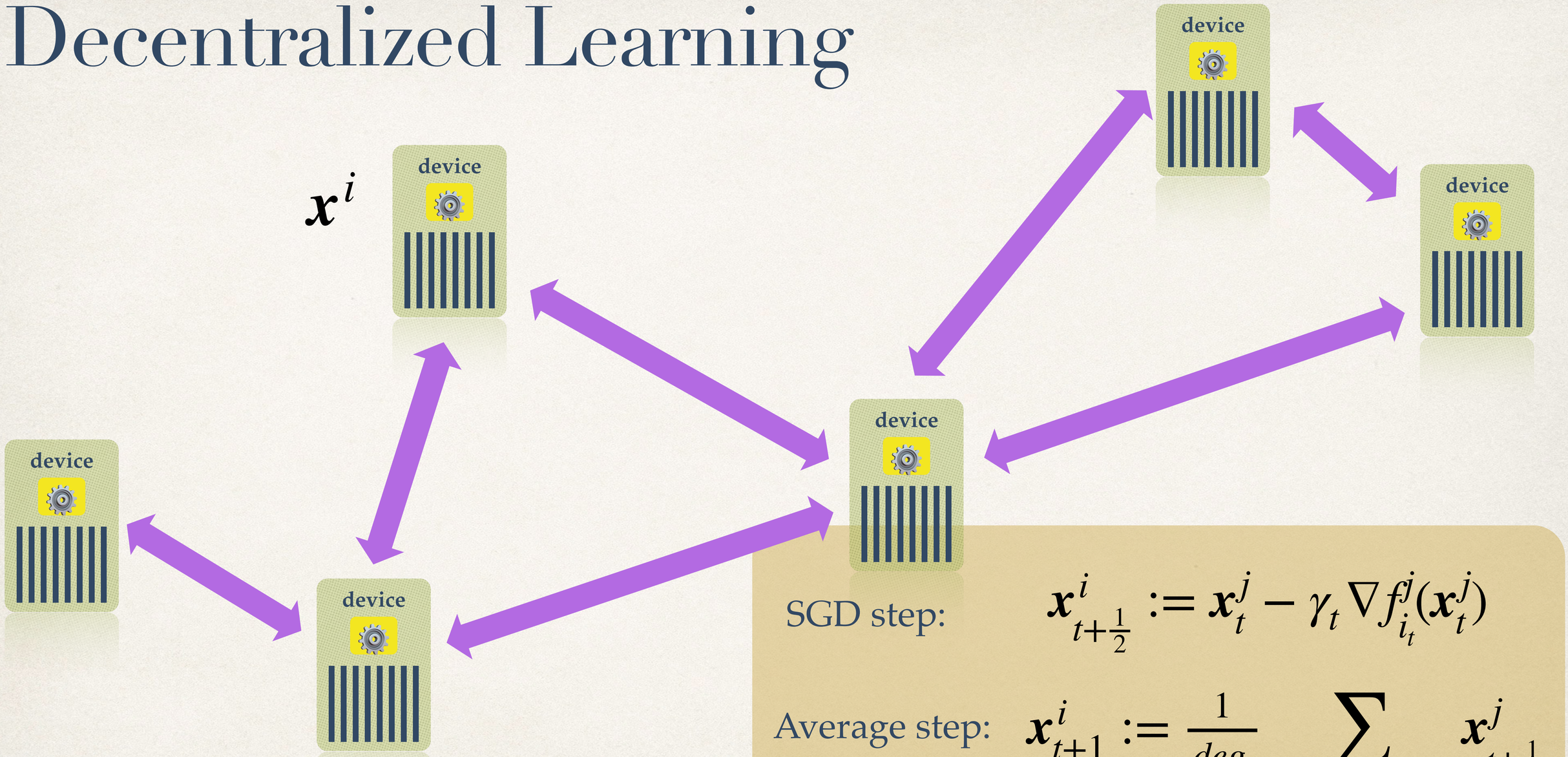


Privacy

Decentralized Learning



Decentralized Learning



SGD step: $$x^i_{t+\frac{1}{2}} := x^j_t - \gamma_t \nabla f^j_{i_t}(x^j_t)$$

Average step: $$x^i_{t+1} := \frac{1}{deg_i} \sum_{j:neighbours} x^j_{t+\frac{1}{2}}$$

Communication Compression

- limited-bit precision vector

  e.g. 1-bit per entry reduces communication 32 times

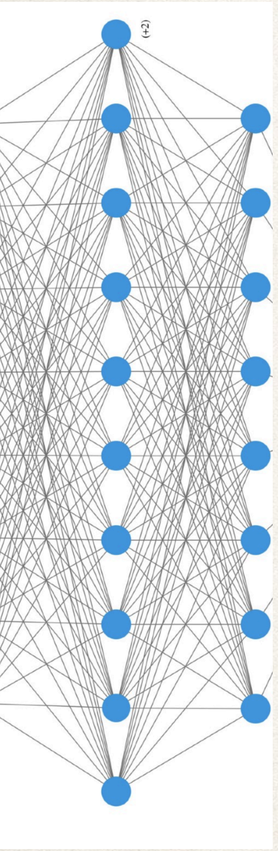
- random/top k% of all the entries

  e.g. k=0.1% reduces communication 1000 times

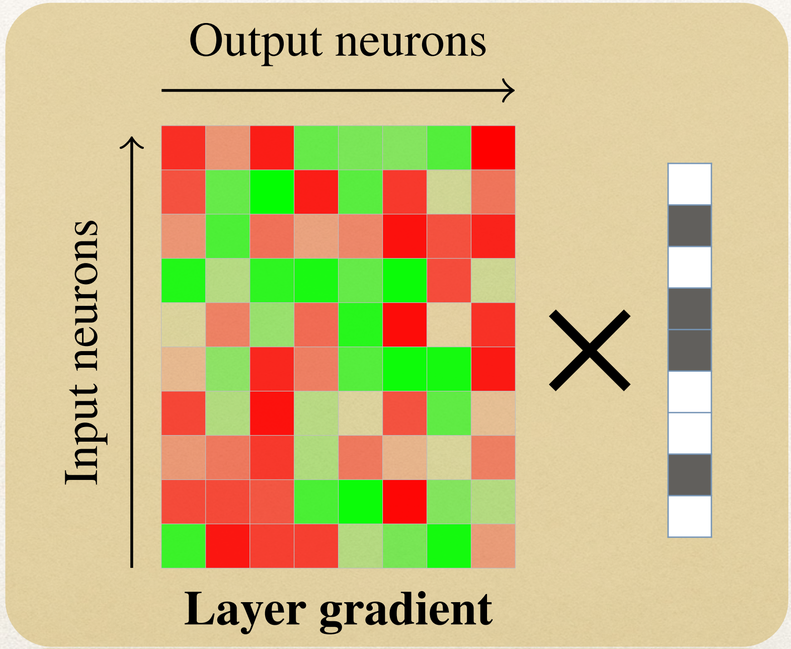
- low rank version of the gradient?

Low-Rank Communication Compression

- PowerSGD



backprop is fast:
linear time



fast compression?

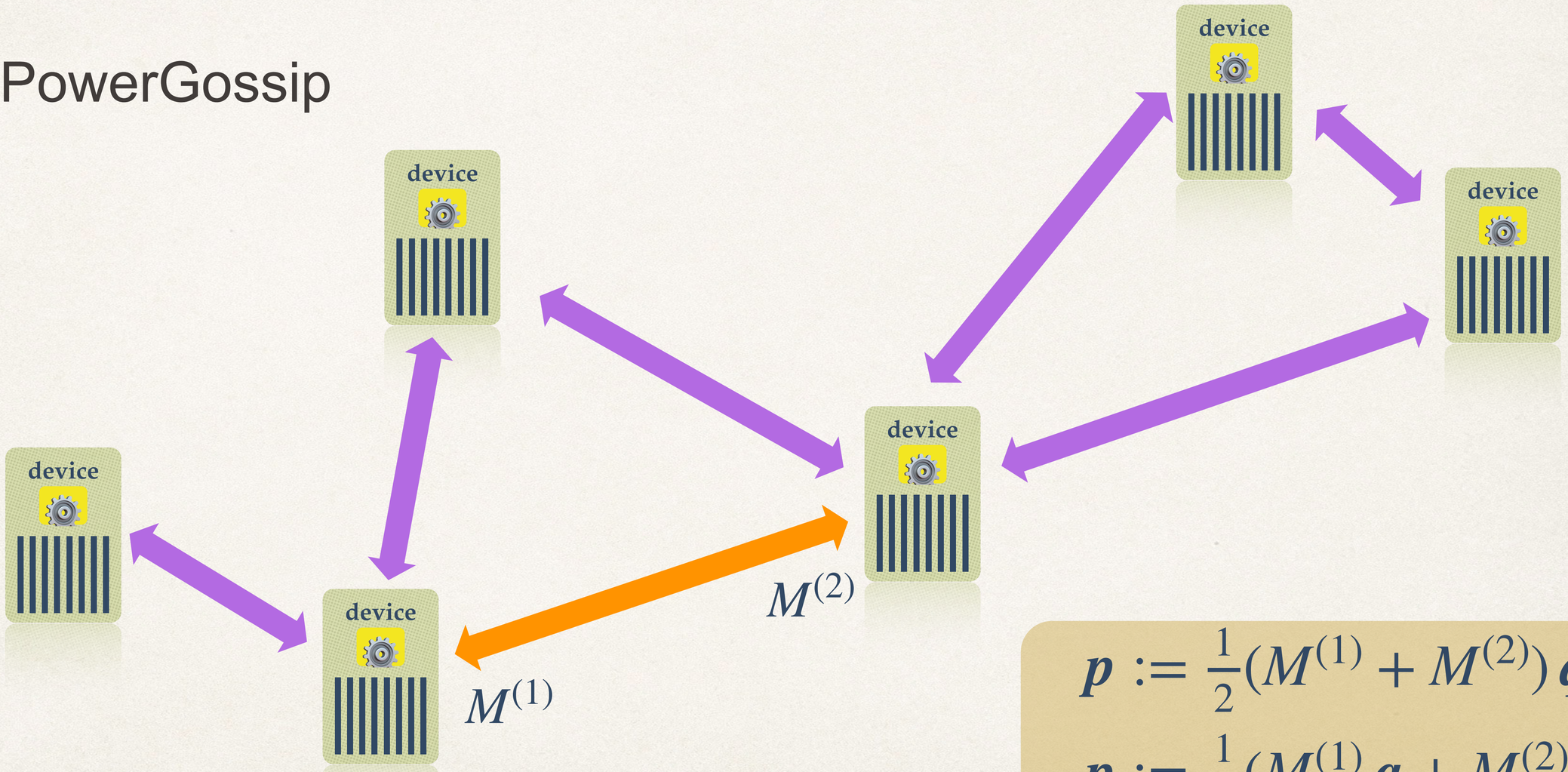
Fast power iterations

$$p := Gq$$
$$q := G^\top p$$

$$\mathscr{C}(G) = pq^\top$$

Decentralized Learning with Compression

- PowerGossip



$$p := \frac{1}{2}(M^{(1)} + M^{(2)})\, q$$

$$p := \frac{1}{2}(M^{(1)}\, q + M^{(2)}\, q)$$

Building Blocks for Decentralized ML

- **Efficiency: Communication & Compute**

  on-device learning, Edge AI

  peer-to-peer communication

- **Privacy**

  data locality, leakage?, attacks?

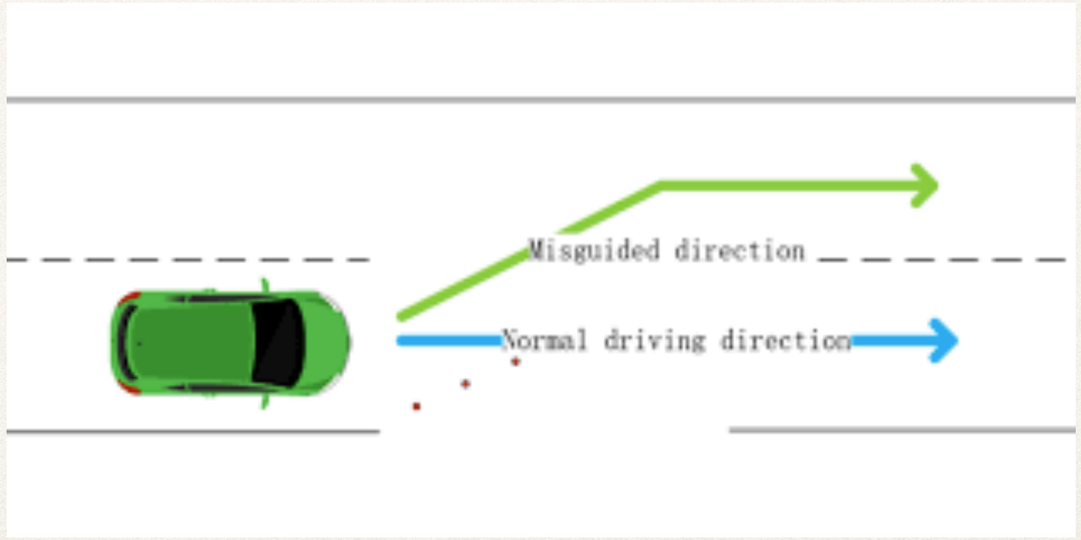
- **Robustness & Incentives**

  tolerate bad players, reward collaboration

3

Robustness

During Training and Inference

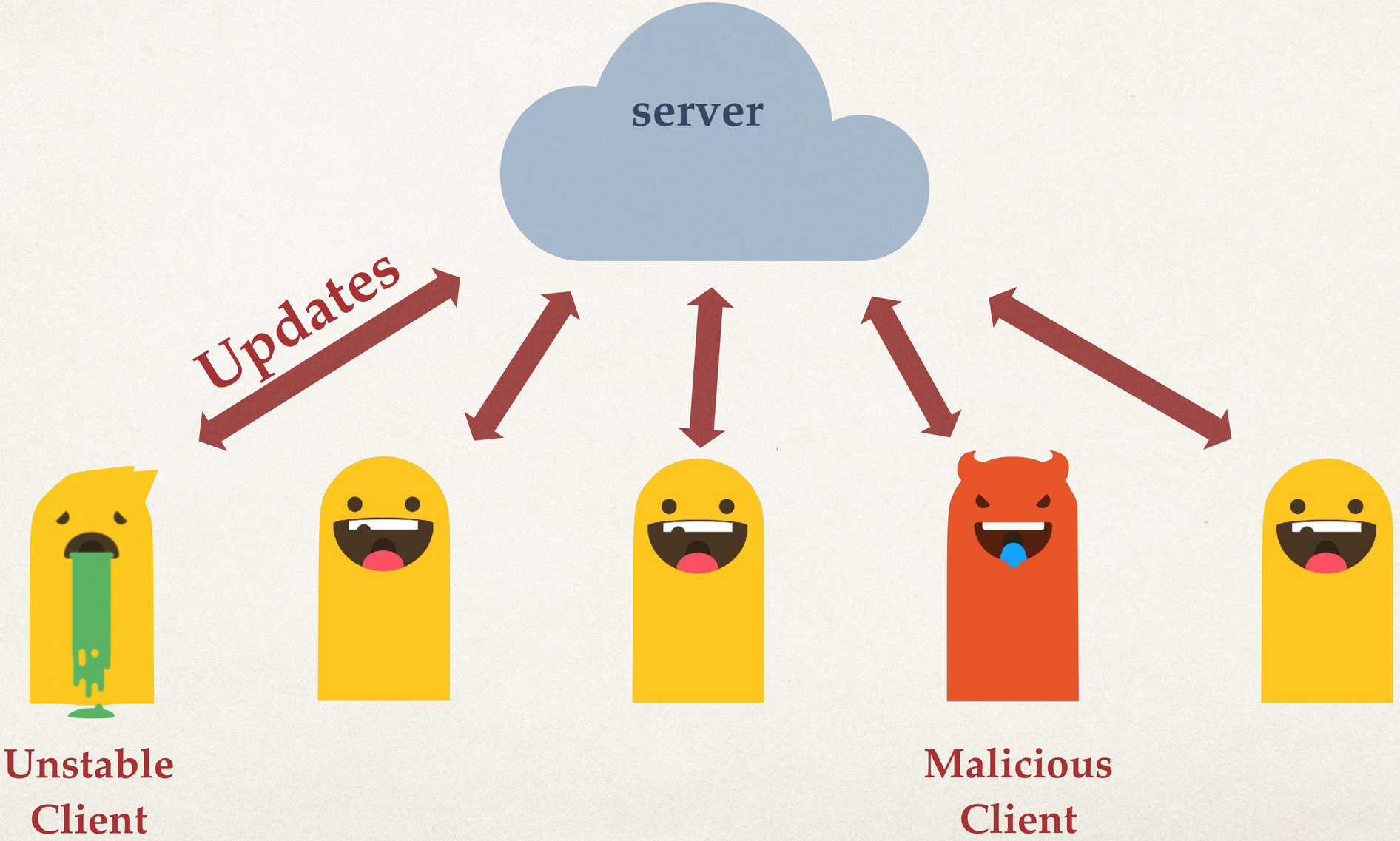


3a

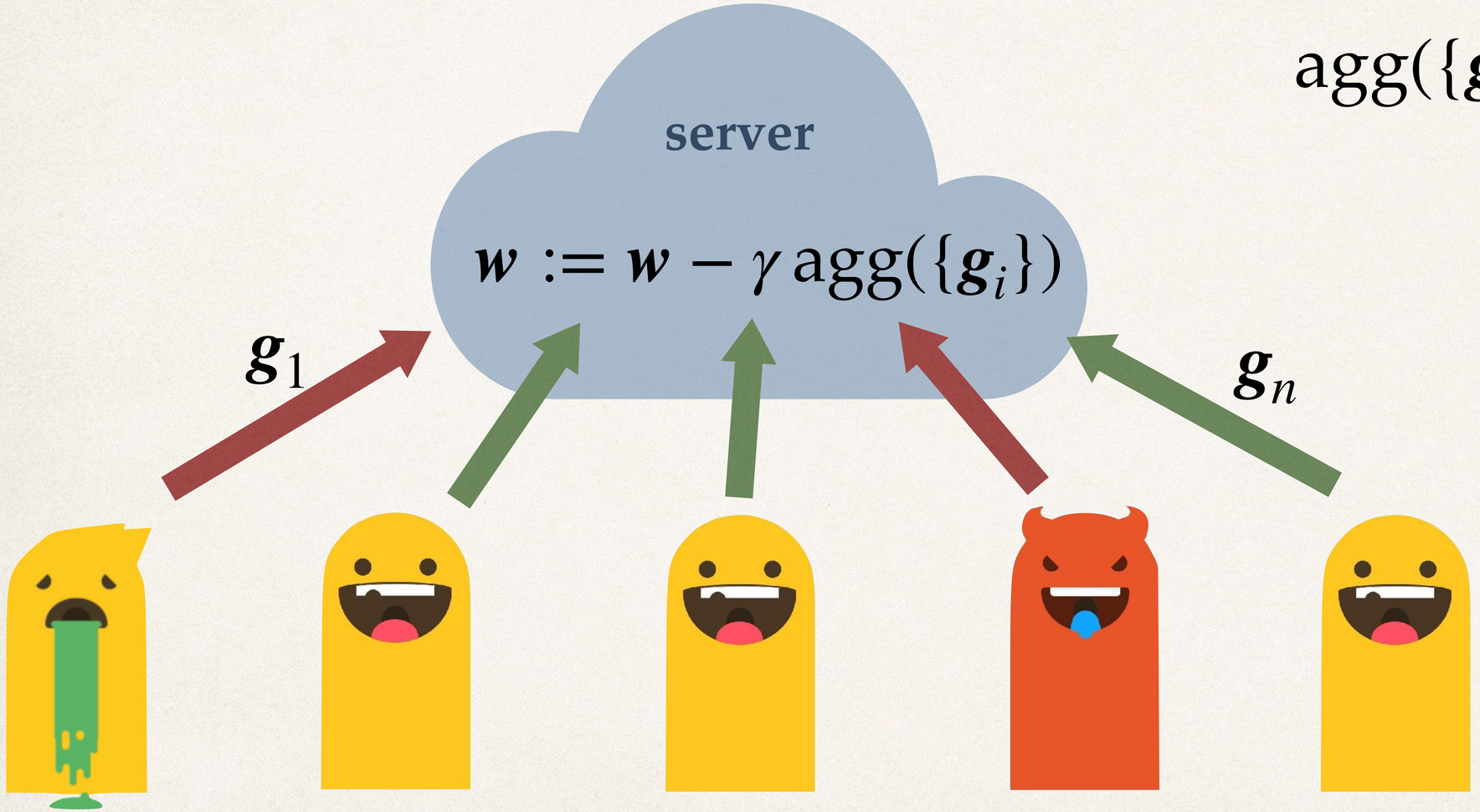
Gradients from faulty/malicious collaborators:

- Byzantine-robust Training

Malicious actors in FL



Byzantine Robust Training



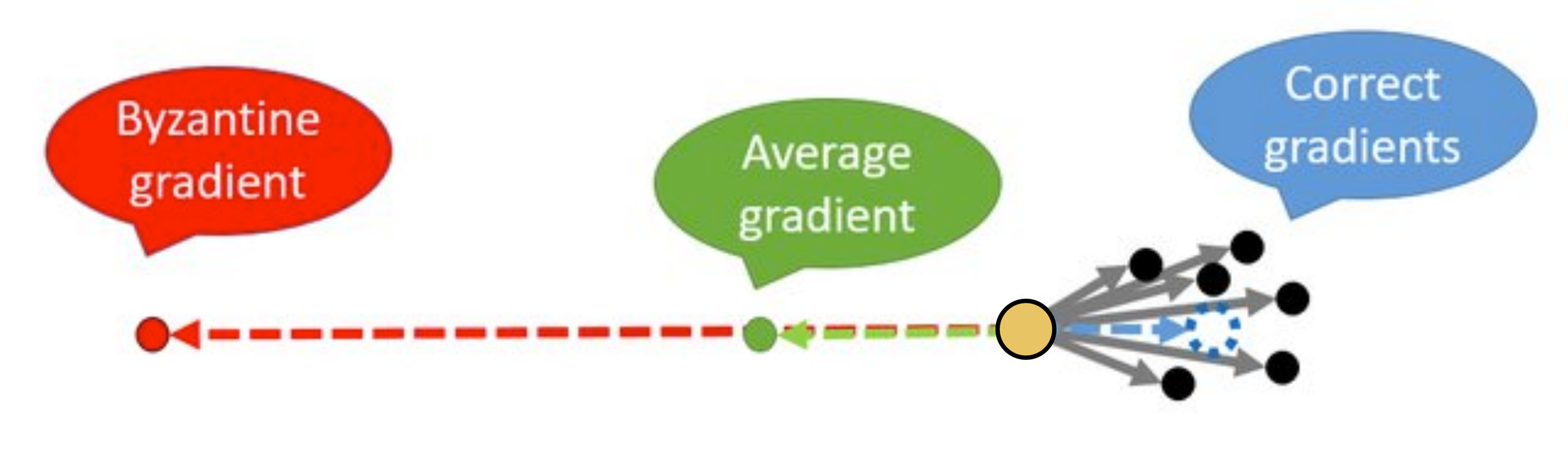
$$\mathrm{agg}(\{g_i\}) := \mathrm{avg}(\{g_i\})$$

$$:= \mathrm{CM}(\{g_i\})$$

Examples:

- Coordinate-wise median [Yin et al. 2017]
- Krum [Blanchard et al. 2018]
- Geometric median / RFA [Pillutla et al. 2019]

Byzantine-robust training



- **Mean vs median**

Zeno: Byzantine-suspicious stochastic gradient descent, 2018, Cong Xie et al.

Negative result

- Robustness of the aggregation rule $\mathrm{agg}(\{\boldsymbol{g}_i\})$
  does **not** imply robust training:
  time-coupled attacks - "little is enough"

- Any aggregation rule which does not use history
  can **fail** for training (convergence)

Fix: Using history with momentum

- Simply use worker momentum

$$\boldsymbol{m}_i := (1-\beta)\boldsymbol{g}_i + \beta \boldsymbol{m}_i$$

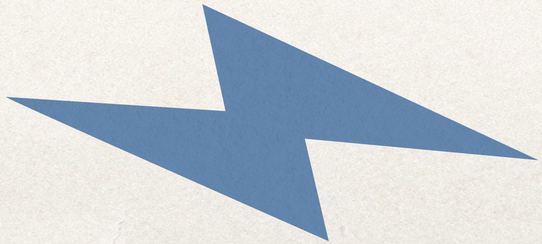
- Effectively averages past gradients, reducing variance

- (Robustly) aggregate worker momentum instead of gradients

$$\boldsymbol{w} := \boldsymbol{w} - \gamma \operatorname{agg}(\{\boldsymbol{m}_i\})$$

Robustness vs Fairness

	Robust mean	Federated	Fairness
Objective	$\text{robust-mean}_i \; f_i(\boldsymbol{x})$ $= \frac{1}{\lvert good \rvert} \sum_{i \in good} f_i(\boldsymbol{x})$	$\frac{1}{n} \sum_{i}^{n} f_i(\boldsymbol{x})$	$\max_i f_i(\boldsymbol{x})$



3b Adversarial Attacks (at inference time)

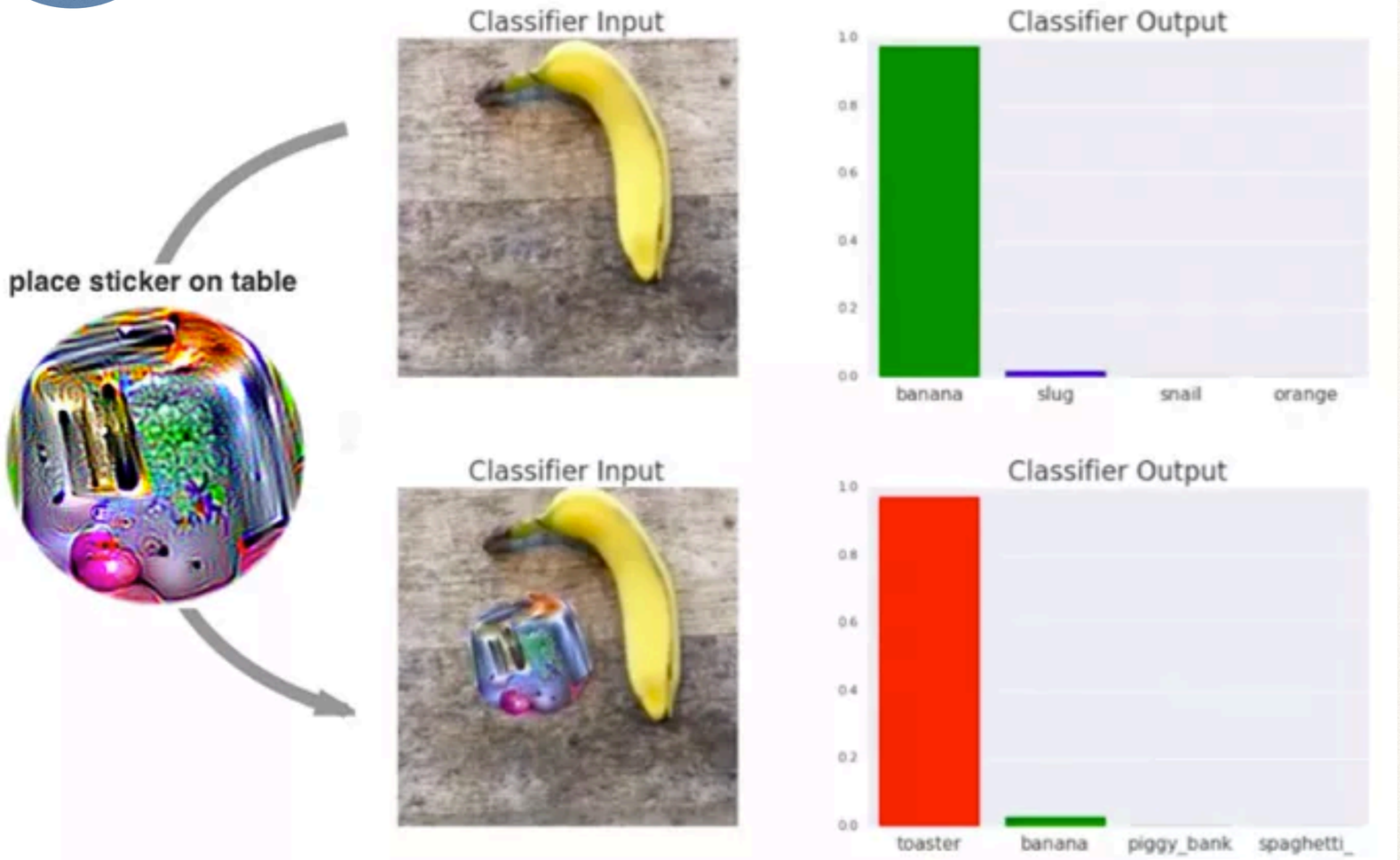


Image: [Tom B. Brown/Dandelion Mané]

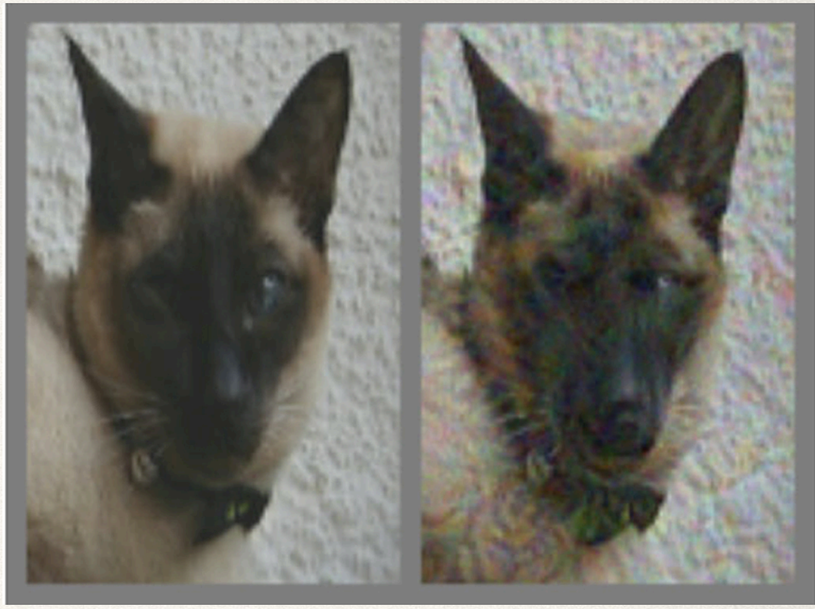


Image: Elsayed ,Papernot et al 2018

Adversarial Attacks (at inference time)

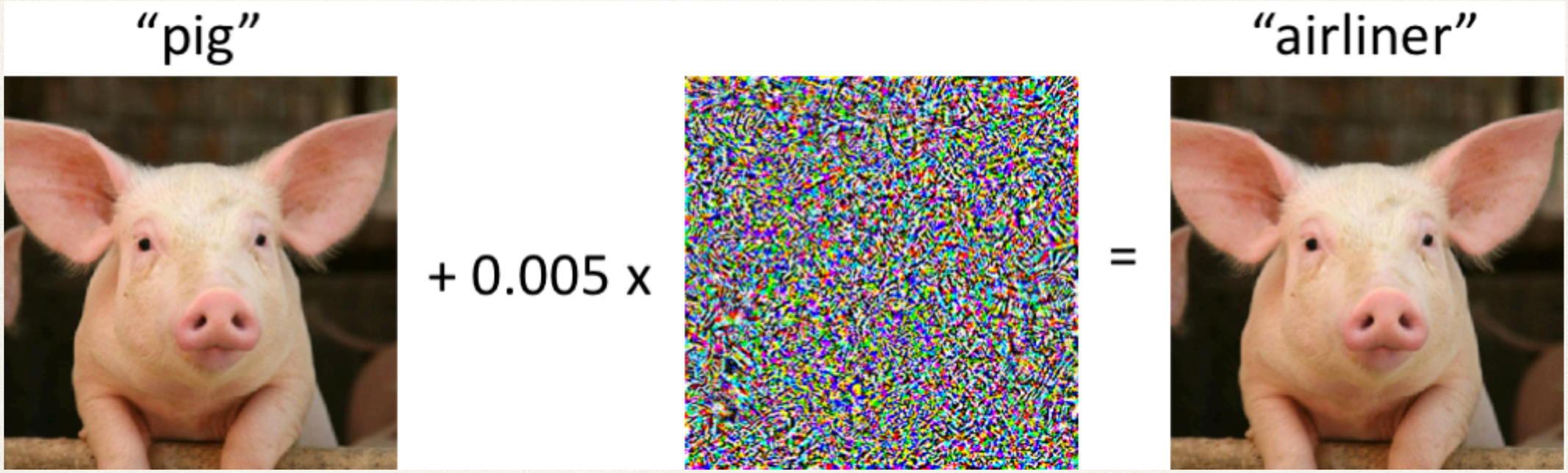


Image: Mądry, Schmidt

More info:
http://gradientscience.org/intro_adversarial/

Adversarial Attacks

- Standard **training**

$$\min_{\mathbf{w}} \quad f_{\mathbf{w}}(\mathbf{x}_i)$$

$\nabla_{\boldsymbol{w}} f$

change **model**

- **Attacking**

$$\max_{\mathbf{x} \in R_\infty(\mathbf{x}_i, \varepsilon)} \quad f_{\mathbf{w}}(\mathbf{x}_i)$$

$\nabla_{\boldsymbol{x}_i} f$

change **data**

- by **Projected Gradient Descent!**

4

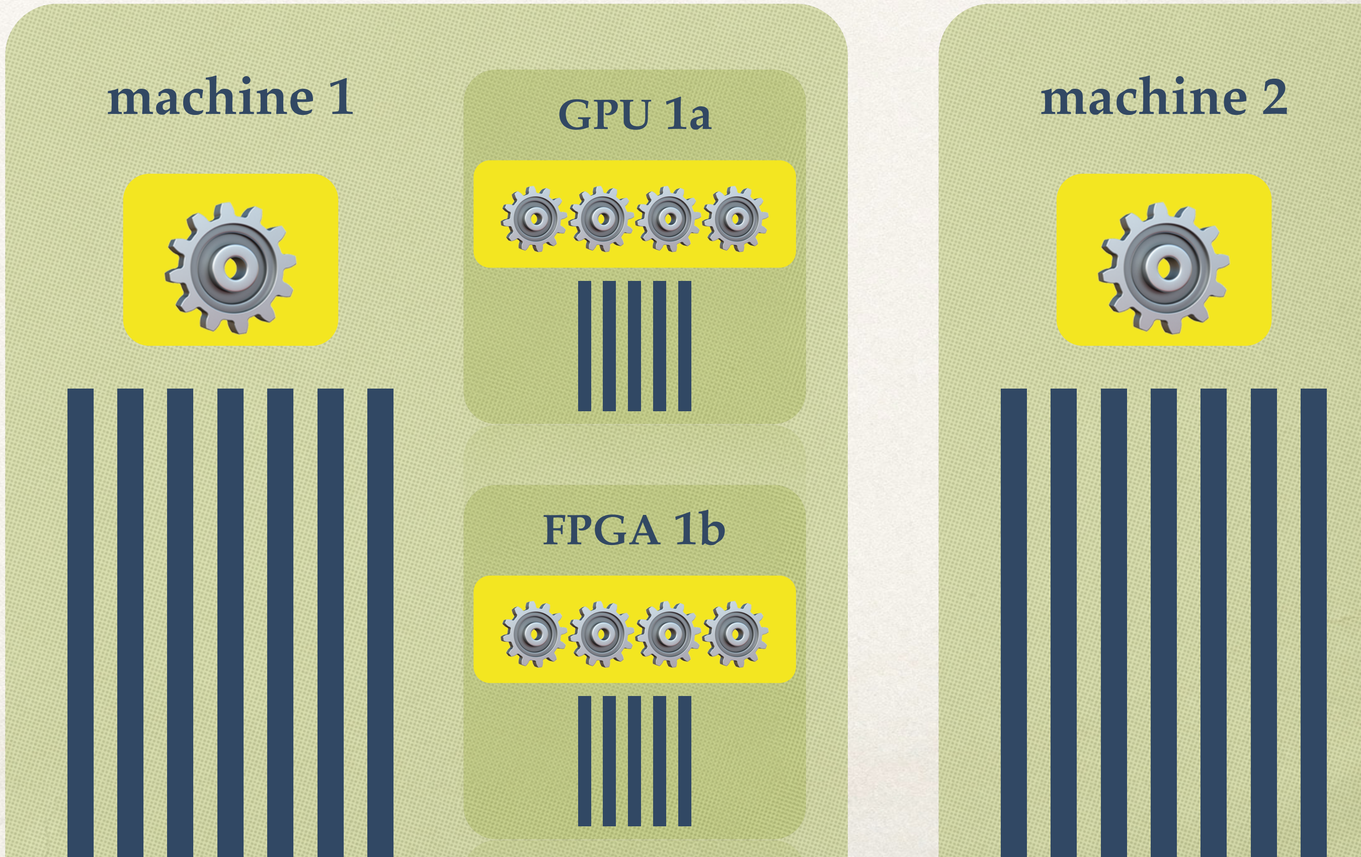
Privacy

- Secure Multiparty Computation
  - secure aggregation
    (private gradients, public model)
- Differential Privacy
- Privacy / inference Attacks

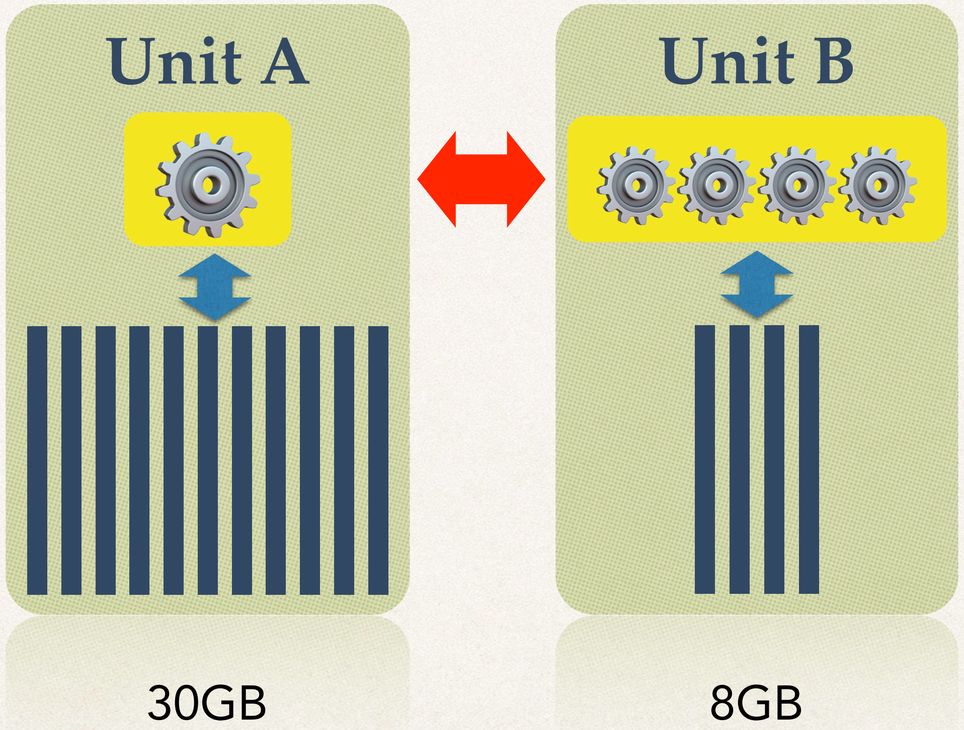
5

Leveraging Heterogenous Systems

Compute & Memory Hierarchy: Which data to put in which device?



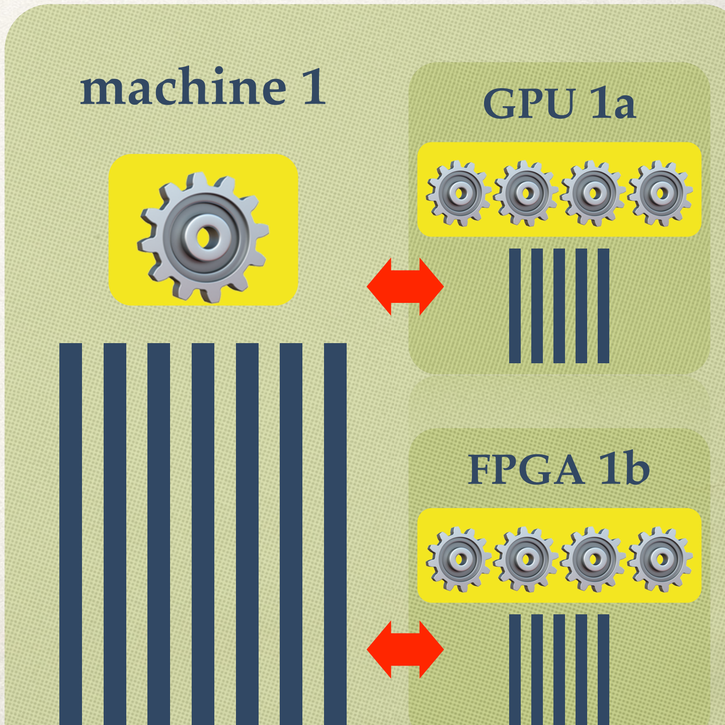
Leveraging Heterogenous Systems



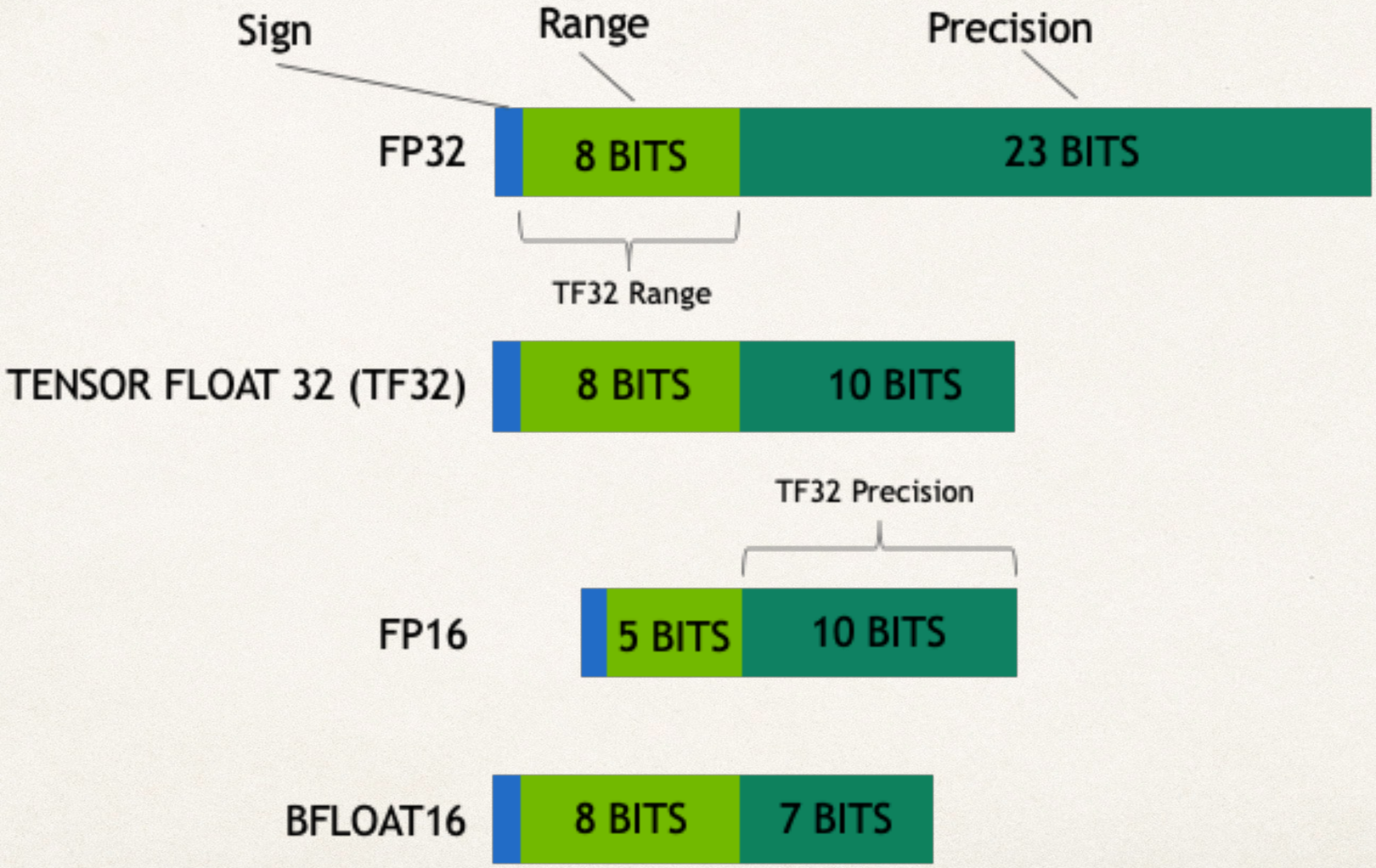
adaptive importance sampling of datapoint
e.g. for general linear models, or word2vec

Trends - Systems

- new **hardware**
  - TPU, GraphCore, Cerebras
  - sparse ops
  - efficient numerics (limited precision), model compression
- **Software** frameworks
  - AutoGrad (Jax, PyTorch, TensorFlow etc)
  - Backends for new hardware



Number formats for DL



Thanks!

mlo.epfl.ch