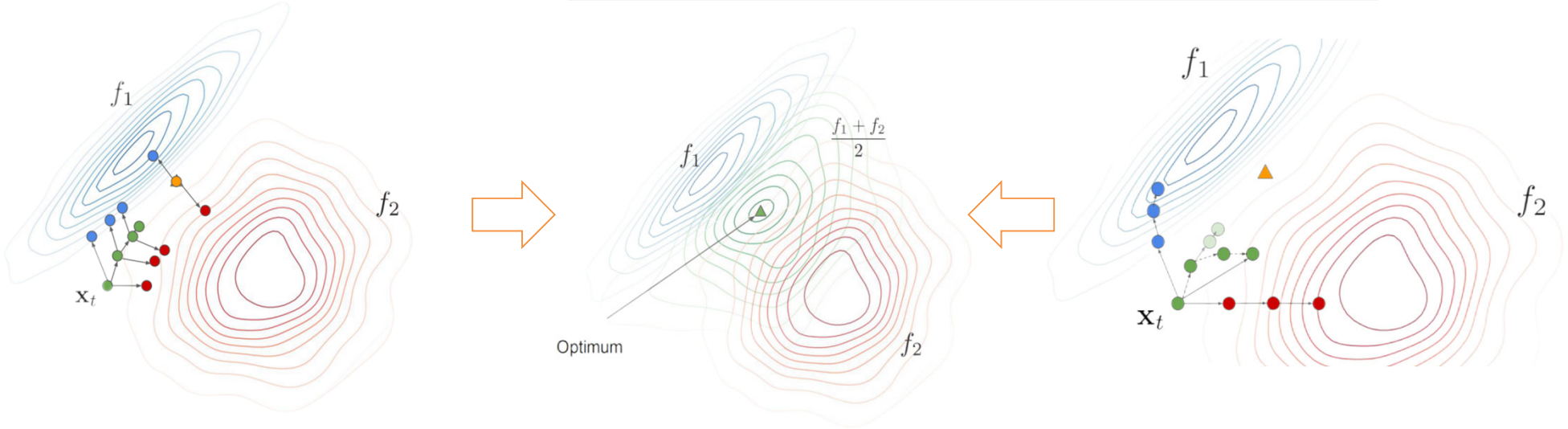
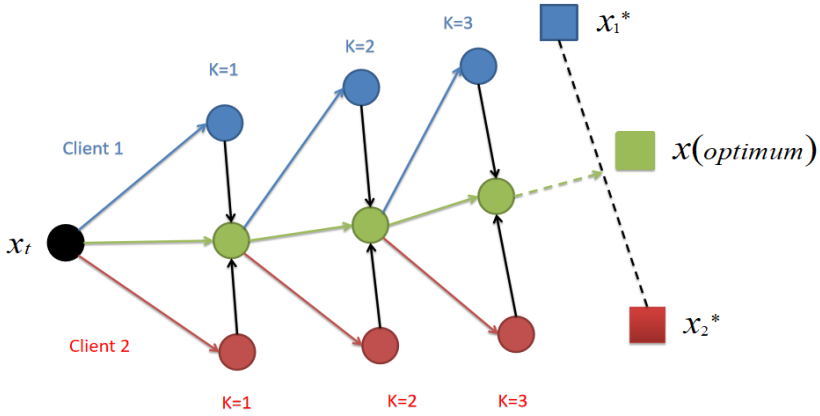
SCAFFOLD: Stochastic Controlled Averaging for Federated Learning

Sai Praneeth Karimireddy [1 2] Satyen Kale [3] Mehryar Mohri [3 4] Sashank J. Reddi [3] Sebastian U. Stich [1]
Ananda Theertha Suresh [3]

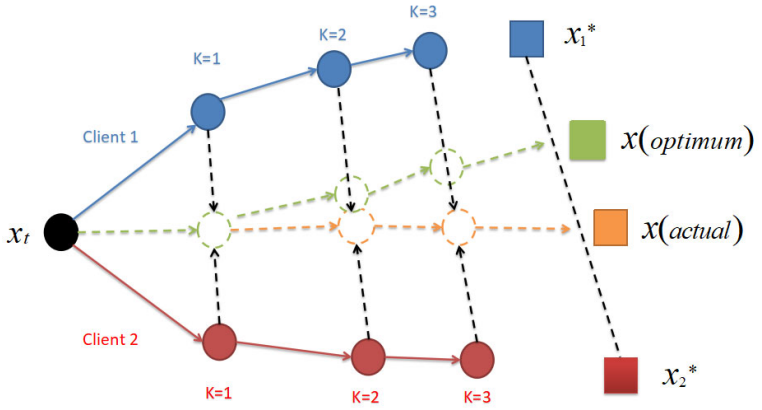
ICML 2020

Motivation (Convergence of FedAvg)





SGD



FedAvg

Method (SGD+FedAvg)

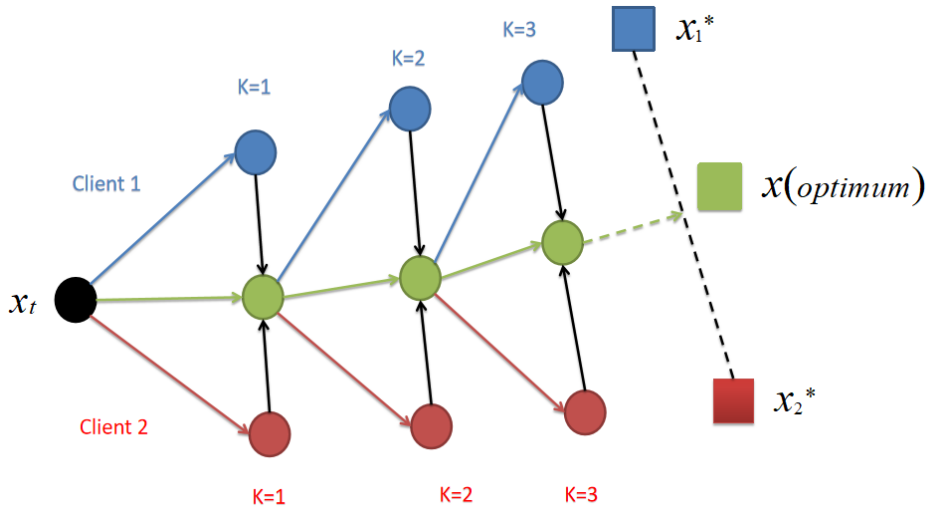
1. How to accelerate global model convergence while ensuring low communication?

① SGD(Good convergence speed+Bad communication overheads)

② FedAvg (low communication overheads+Bad convergence speed)

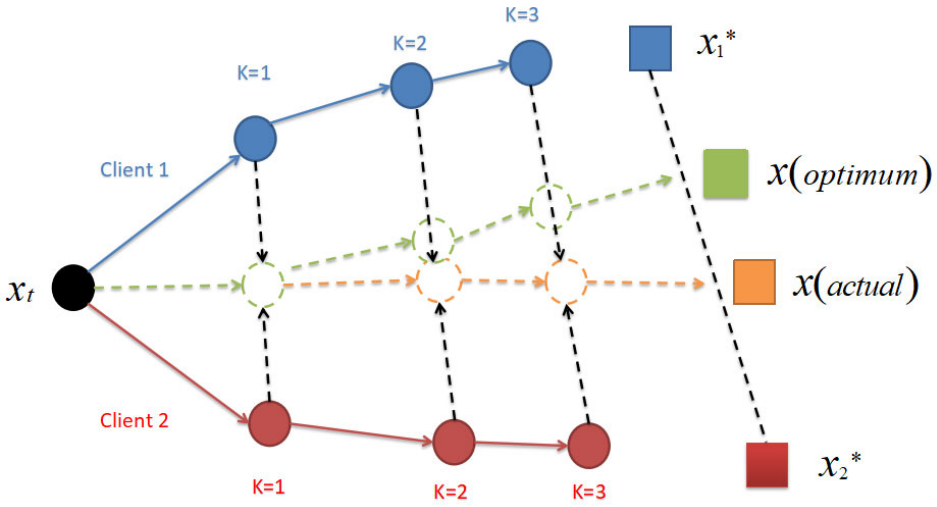
A new algorithm with good convergence speed and low communication overheads

2. SGD features in updating the global model

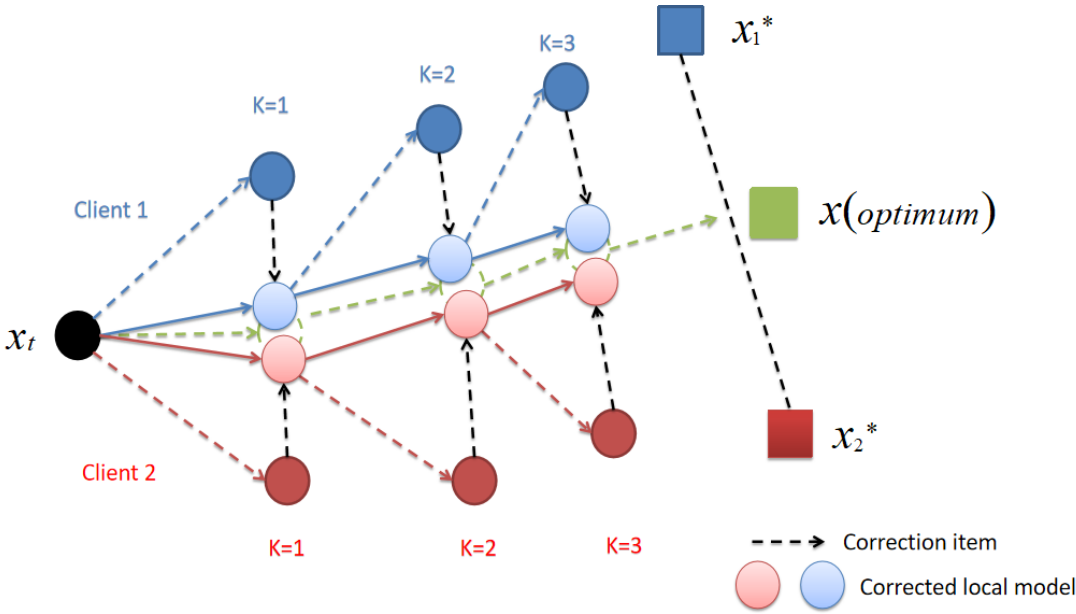


Method (SGD+FedAvg)

3. How to add features from SGD to FedAvg?



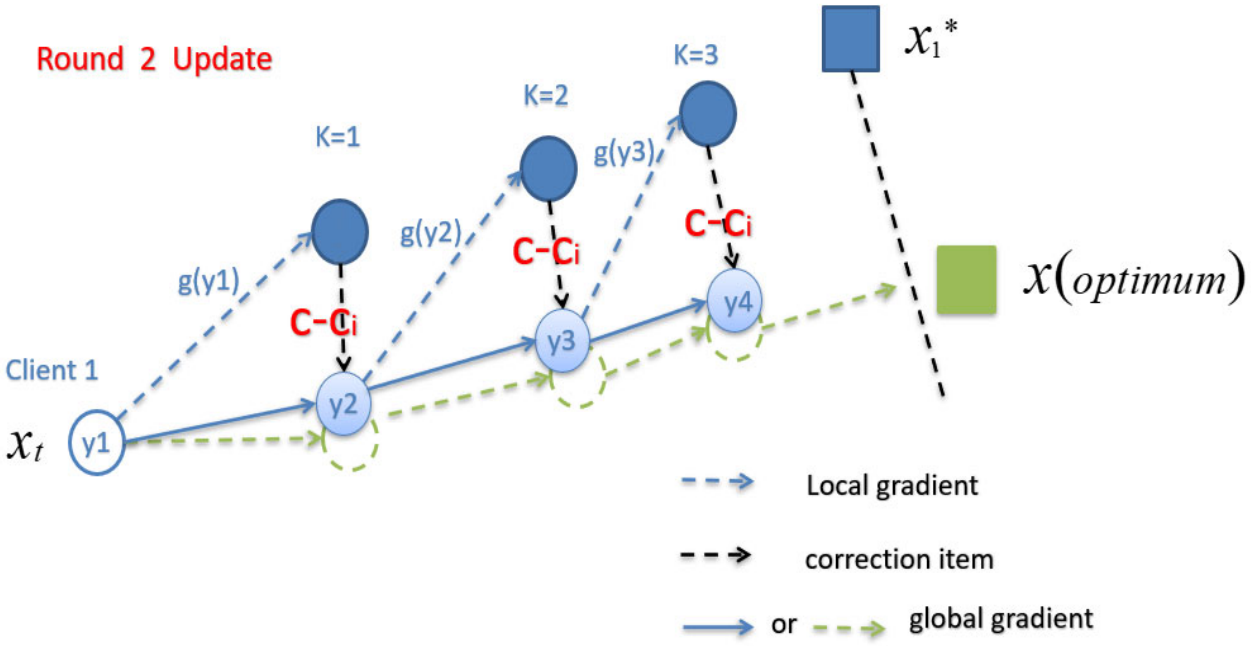
original FedAvg algorithm



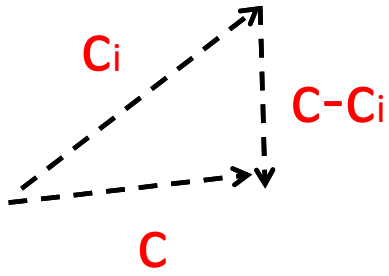
Improved FedAvg algorithm(SCAFFOLD)

Method (SGD+FedAvg)

4. How to find correction item?



Round 1 update



Ci: average gradient of client 1 local k updates in Round 1.

C: global average gradient, The first round of updates is aggregated (Ci) and the second round is sent to each client

$$c = \frac{1}{N}\sum c_i$$

Method (SGD+FedAvg)

5. Two ways to update client model average gradient

$$\boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \eta_l(g_i(\boldsymbol{y}_i) + \boldsymbol{c} - \boldsymbol{c}_i)\,.$$

$$\boldsymbol{c}_i^+ \leftarrow \begin{cases} \text{Option I.} & g_i(\boldsymbol{x})\,, \text{or} \\ \text{Option II.} & \boldsymbol{c}_i - \boldsymbol{c} + \frac{1}{K\eta_l}(\boldsymbol{x} - \boldsymbol{y}_i)\,. \end{cases}$$

option I. Calculate the client gradient using the initial global model x(provided by the server)

SCAFFOLD and FedAvg

Algorithm 2 FEDAVG: Federated Averaging

server input: initial $\boldsymbol{x}$, and global step-size η_g
client i's input: local step-size η_l
for each round $r = 1, \ldots, R$ **do**
  sample clients $\mathcal{S} \subseteq \{1, \ldots, N\}$
  communicate $\boldsymbol{x}$ to all clients $i \in \mathcal{S}$
  on client $i \in \mathcal{S}$ **in parallel do**
    initialize local model $\boldsymbol{y}_i \leftarrow \boldsymbol{x}$
    for $k = 1, \ldots, K$ **do**
      compute mini-batch gradient $g_i(\boldsymbol{y}_i)$
      $\boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \eta_l g_i(\boldsymbol{y}_i)$
    end for
    communicate $\Delta \boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \boldsymbol{x}$
  end on client
  $\Delta \boldsymbol{x} \leftarrow \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \Delta \boldsymbol{y}_i$
  $\boldsymbol{x} \leftarrow \boldsymbol{x} + \eta_g \Delta \boldsymbol{x}$
end for

Algorithm 1 SCAFFOLD: Stochastic Controlled Averaging for federated learning

server input: initial $\boldsymbol{x}$ and $\boldsymbol{c}$, and global step-size η_g
client i's input: $\boldsymbol{c}_i$, and local step-size η_l
for each round $r = 1, \ldots, R$ **do**
  sample clients $\mathcal{S} \subseteq \{1, \ldots, N\}$
  communicate $(\boldsymbol{x}, \boldsymbol{c})$ to all clients $i \in \mathcal{S}$
  on client $i \in \mathcal{S}$ **in parallel do**
    initialize local model $\boldsymbol{y}_i \leftarrow \boldsymbol{x}$
    for $k = 1, \ldots, K$ **do**
      compute mini-batch gradient $g_i(\boldsymbol{y}_i)$
      $\boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \eta_l \left(g_i(\boldsymbol{y}_i) - \boldsymbol{c}_i + \boldsymbol{c}\right)$
    end for
    $\boldsymbol{c}_i^+ \leftarrow$ (i) $g_i(\boldsymbol{x})$, or (ii) $\boldsymbol{c}_i - \boldsymbol{c} + \frac{1}{K\eta_l}(\boldsymbol{x} - \boldsymbol{y}_i)$
    communicate $(\Delta \boldsymbol{y}_i, \Delta \boldsymbol{c}_i) \leftarrow (\boldsymbol{y}_i - \boldsymbol{x}, \boldsymbol{c}_i^+ - \boldsymbol{c}_i)$
    $\boldsymbol{c}_i \leftarrow \boldsymbol{c}_i^+$
  end on client
  $(\Delta \boldsymbol{x}, \Delta \boldsymbol{c}) \leftarrow \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} (\Delta \boldsymbol{y}_i, \Delta \boldsymbol{c}_i)$
  $\boldsymbol{x} \leftarrow \boldsymbol{x} + \eta_g \Delta \boldsymbol{x}$ and $\boldsymbol{c} \leftarrow \boldsymbol{c} + \frac{|\mathcal{S}|}{N} \Delta \boldsymbol{c}$
end for

Experiments

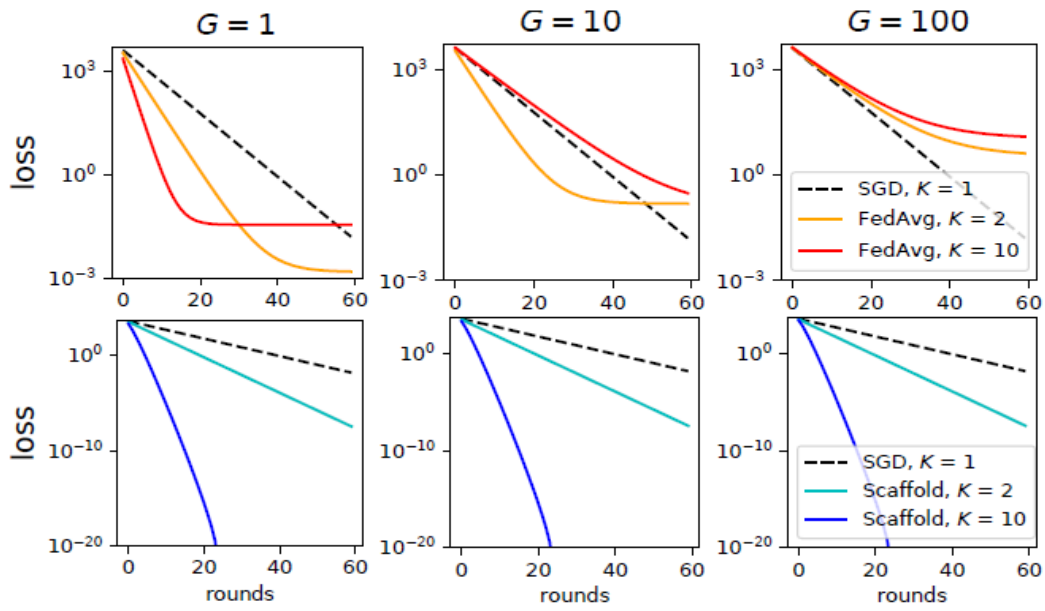


Figure 3. SGD (dashed black), FedAvg (above), and SCAFFOLD (below) on simulated data. FedAvg gets worse as local steps increases with $K = 10$ (red) worse than $K = 2$ (orange). It also gets slower as gradient-dissimilarity (G) increases (to the right). SCAFFOLD significantly improves with more local steps, with $K = 10$ (blue) faster than $K = 2$ (light blue) and SGD. Its performance is identical as we vary heterogeneity (G).

(A1) (G, B)**-BGD** or bounded gradient dissimilarity: there exist constants $G \geq 0$ and $B \geq 1$ such that

$$\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{x})\|^2 \leq G^2 + B^2\|\nabla f(\boldsymbol{x})\|^2\,,\ \forall \boldsymbol{x}\,.$$

If $\{f_i\}$ are convex, we can relax the assumption to

$$\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{x})\|^2 \leq G^2 + 2\beta B^2(f(\boldsymbol{x}) - f^\star)\,,\ \forall \boldsymbol{x}\,.$$

Experiments

Table 3. Communication rounds to reach 0.5 test accuracy for logistic regression on EMNIST as we vary number of epochs. 1k+ indicates 0.5 accuracy was not reached even after 1k rounds, and similarly an arrowhead indicates that the barplot extends beyond the table. 1 epoch for local update methods corresponds to 5 local steps (0.2 batch size), and 20% of clients are sampled each round.

	Epochs	0% similarity (sorted)		10% similarity		100% similarity (i.i.d.)	
		Num. of rounds	Speedup	Num. of rounds	Speedup	Num. of rounds	Speedup
SGD	1	317	(1×)	365	(1×)	416	(1×)
SCAFFOLD	1	77	(4.1×)	62	(5.9×)	60	(6.9×)
	5	152	(2.1×)	20	(18.2×)	10	(41.6×)
	10	286	(1.1×)	16	(22.8×)	7	(59.4×)
	20	266	(1.2×)	11	(33.2×)	4	(104×)
FedAvg	1	258	(1.2×)	74	(4.9×)	83	(5×)
	5	428	(0.7×)	34	(10.7×)	10	(41.6×)
	10	711	(0.4×)	25	(14.6×)	6	(69.3×)
	20	1k+	(< 0.3×)	18	(20.3×)	4	(104×)
FedProx	1	1k+	(< 0.3×)	979	(0.4×)	459	(0.9×)
	5	1k+	(< 0.3×)	794	(0.5×)	351	(1.2×)
	10	1k+	(< 0.3×)	894	(0.4×)	308	(1.4×)
	20	1k+	(< 0.3×)	916	(0.4×)	351	(1.2×)

Experiments

Table 4. Communication rounds to reach 0.45 test accuracy for logistic regression on EMNIST as we vary the number of sampled clients. Number of epochs is kept fixed to 5. SCAFFOLD is consistently faster than FEDAVG. As we decrease the number of clients sampled in each round, the increase in number of rounds is sub-linear. This slow-down is better for more similar clients.

	Clients	0% similarity		10% similarity	
SCAFFOLD	20%	143	(1.0×)	9	(1.0×)
	5%	290	(2.0×)	13	(1.4×)
	1%	790	(5.5×)	28	(3.1×)
FEDAVG	20%	179	(1.0×)	12	(1.0×)
	5%	334	(1.9×)	17	(1.4×)
	1%	1k+	(5.6+×)	35	(2.9×)

THANKS