Byzantine-Resilient High-Dimensional SGD with Local Iterations on Heterogeneous Data

Deepesh Data

University of California, Los Angeles (UCLA), USA

Joint work with **Suhas Diggavi** (UCLA)

ICML, July 18-24, 2021

Motivating Example: Federated Learning (FL)

- Collaborative ML without data centralization [McMahan et al. AISTATS-17, Konecny et al. arXiv-17]
- Building a machine learning model for next word prediction

- Tens of millions of devices
- Different geographic locations

Picture taken from "Machine Learning Blog | ML@CMU"

Motivating Example: Federated Learning (FL)

- Collaborative ML without data centralization [McMahan et al. AISTATS-17, Konecny et al. arXiv-17]
- Building a machine learning model for next word prediction

- **Data heterogeneity**
  - non-iid local data

- Tens of millions of devices
- Different geographic locations

Picture taken from "Machine Learning Blog | ML@CMU"

Motivating Example: Federated Learning (FL)

- Collaborative ML without data centralization [McMahan et al. AISTATS-17, Konecny et al. arXiv-17]
- Building a machine learning model for next word prediction

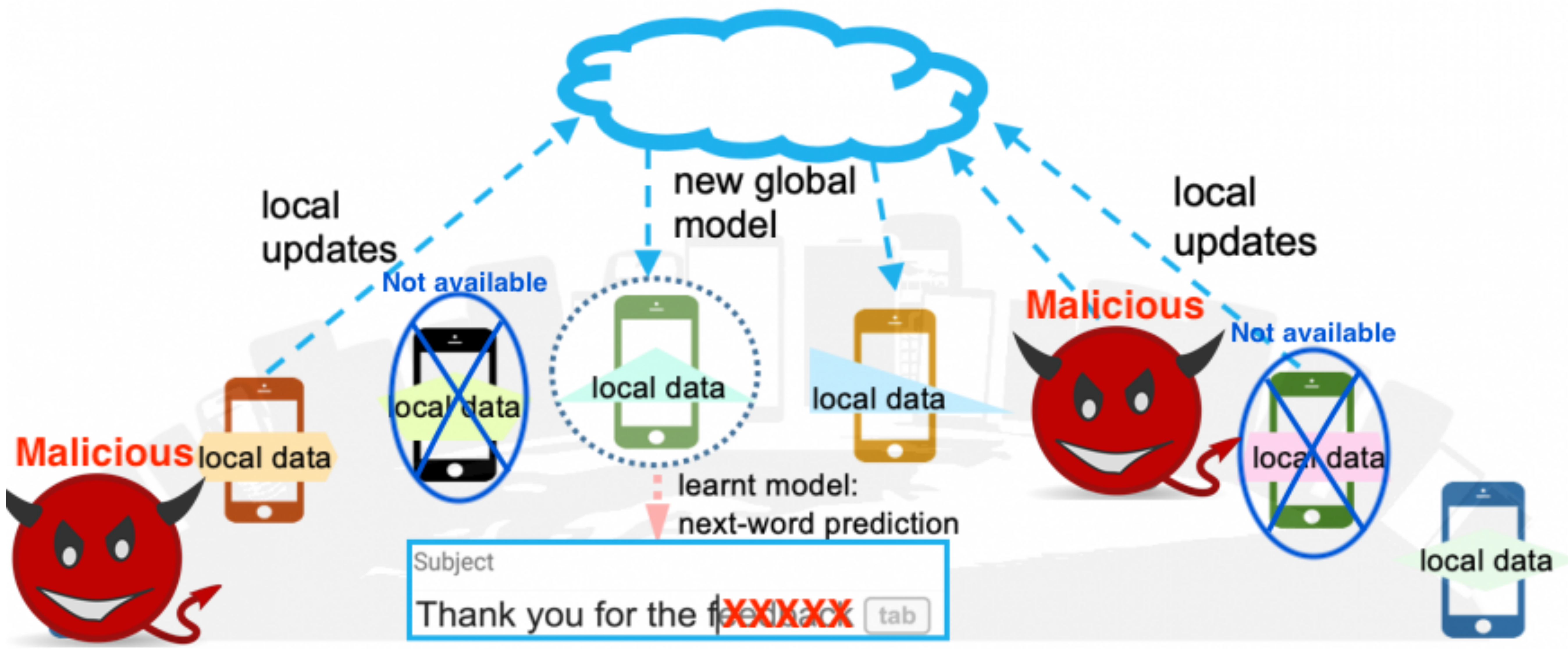
- **Data heterogeneity**
  - non-iid local data
- **Partial device participation**
  - Not all devices are available all the time

- Tens of millions of devices
- Different geographic locations

Picture taken from "Machine Learning Blog | ML@CMU"

Motivating Example: Federated Learning (FL)

- Collaborative ML without data centralization [McMahan et al. AISTATS-17, Konecny et al. arXiv-17]
- Building a machine learning model for next word prediction



- **Data heterogeneity**
  - non-iid local data
- **Partial device participation**
  - Not all devices are available all the time
- **Unreliable/malicious** devices
  - Provide adversarial SGD updates to server

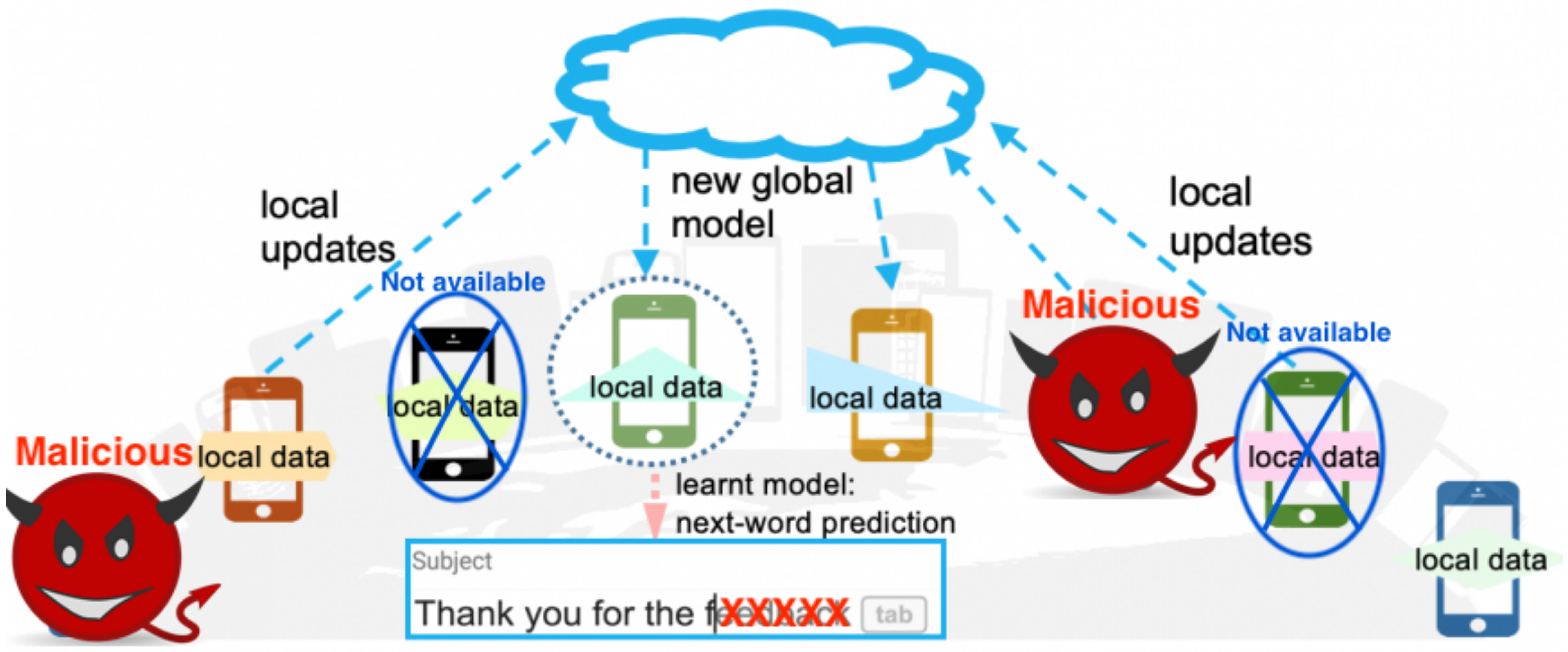
  $\Longrightarrow$ Compromises the accuracy of the software

- Tens of millions of devices
- Different geographic locations

Picture taken from "Machine Learning Blog | ML@CMU"

Motivating Example: Federated Learning (FL)

- Collaborative ML without data centralization [McMahan et al. AISTATS-17, Konecny et al. arXiv-17]
- Building a machine learning model for next word prediction

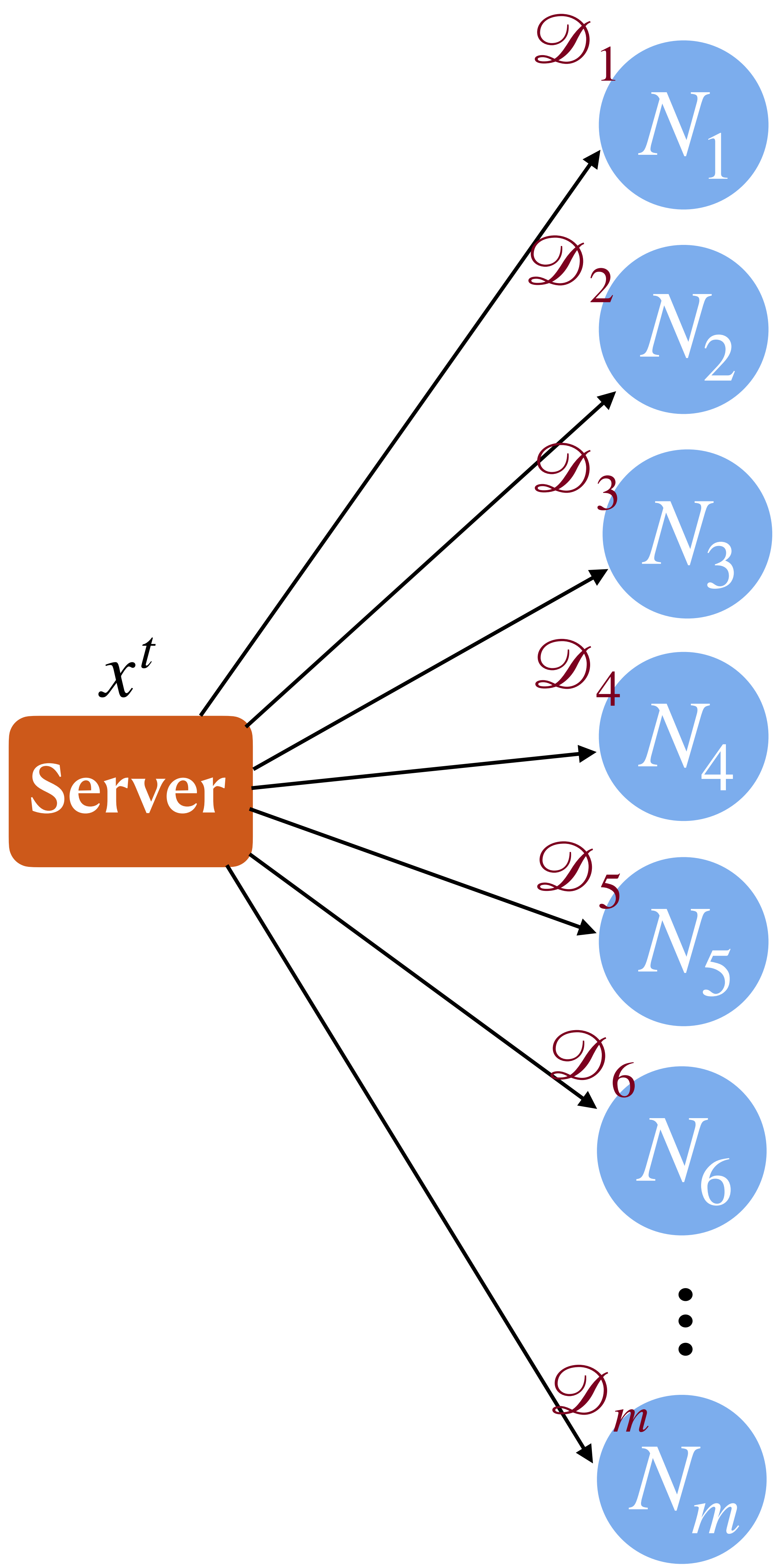


- Tens of millions of devices
- Different geographic locations

- **Data heterogeneity**
  - non-iid local data
- **Partial device participation**
  - Not all devices are available all the time
- **Unreliable/malicious** devices
  - Provide adversarial SGD updates to server
  $\Longrightarrow$ Compromises the accuracy of the software
- **Communication (bandwidth) constraints**
  - Communication in every round is not possible
  $\Longrightarrow$ local iterations

Picture taken from "Machine Learning Blog | ML@CMU"

Distributed Local SGD with Client Sampling

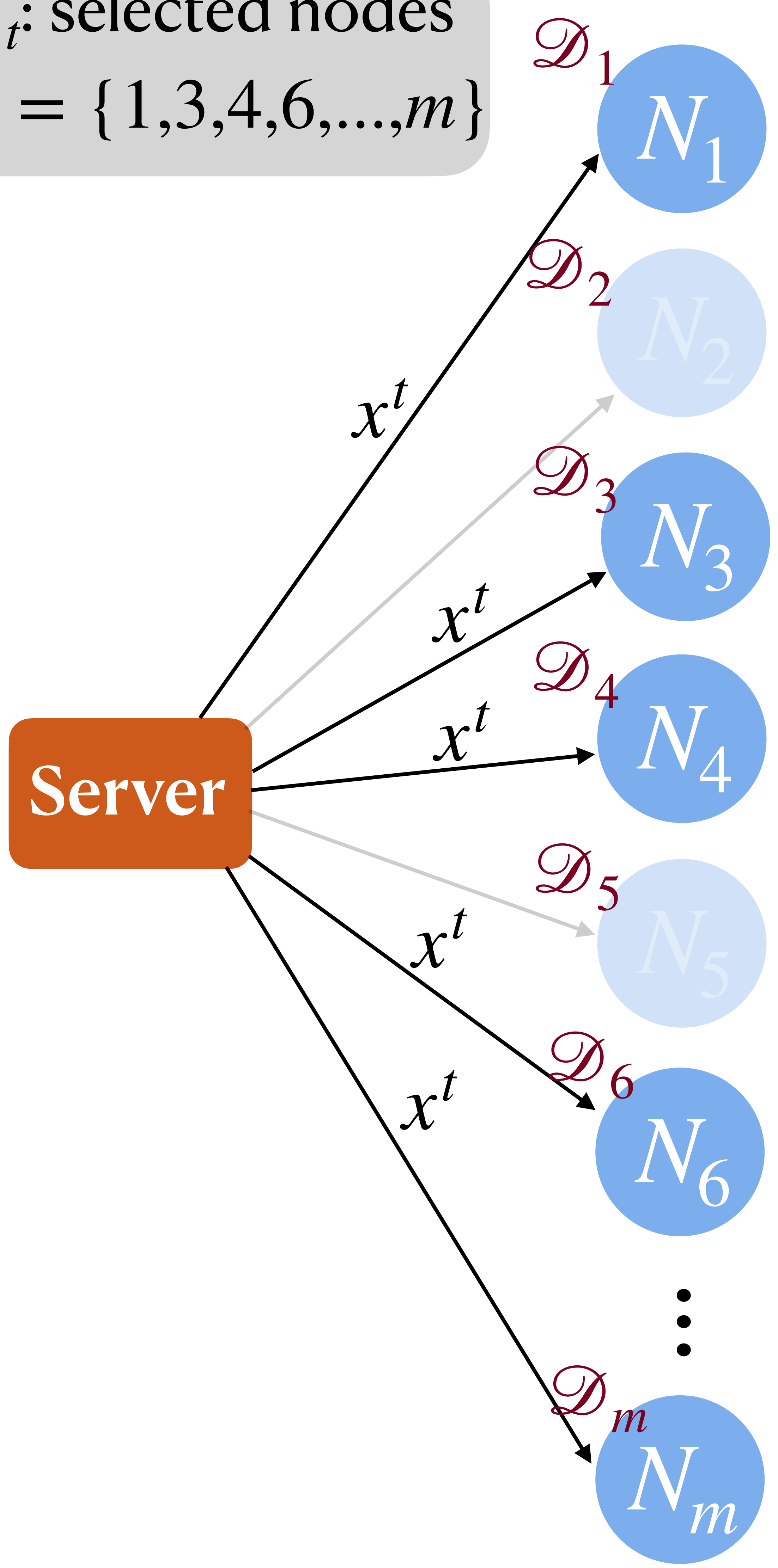


- $\mathscr{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

$$\arg\min_{x\in\mathbb{R}^d}\left(F(x) := \frac{1}{m}\sum_{i=1}^{m} F_i(x)\right)$$

Distributed Local SGD with Client Sampling

$\mathcal{K}_t$: selected nodes
$\mathcal{K}_t = \{1,3,4,6,\ldots,m\}$

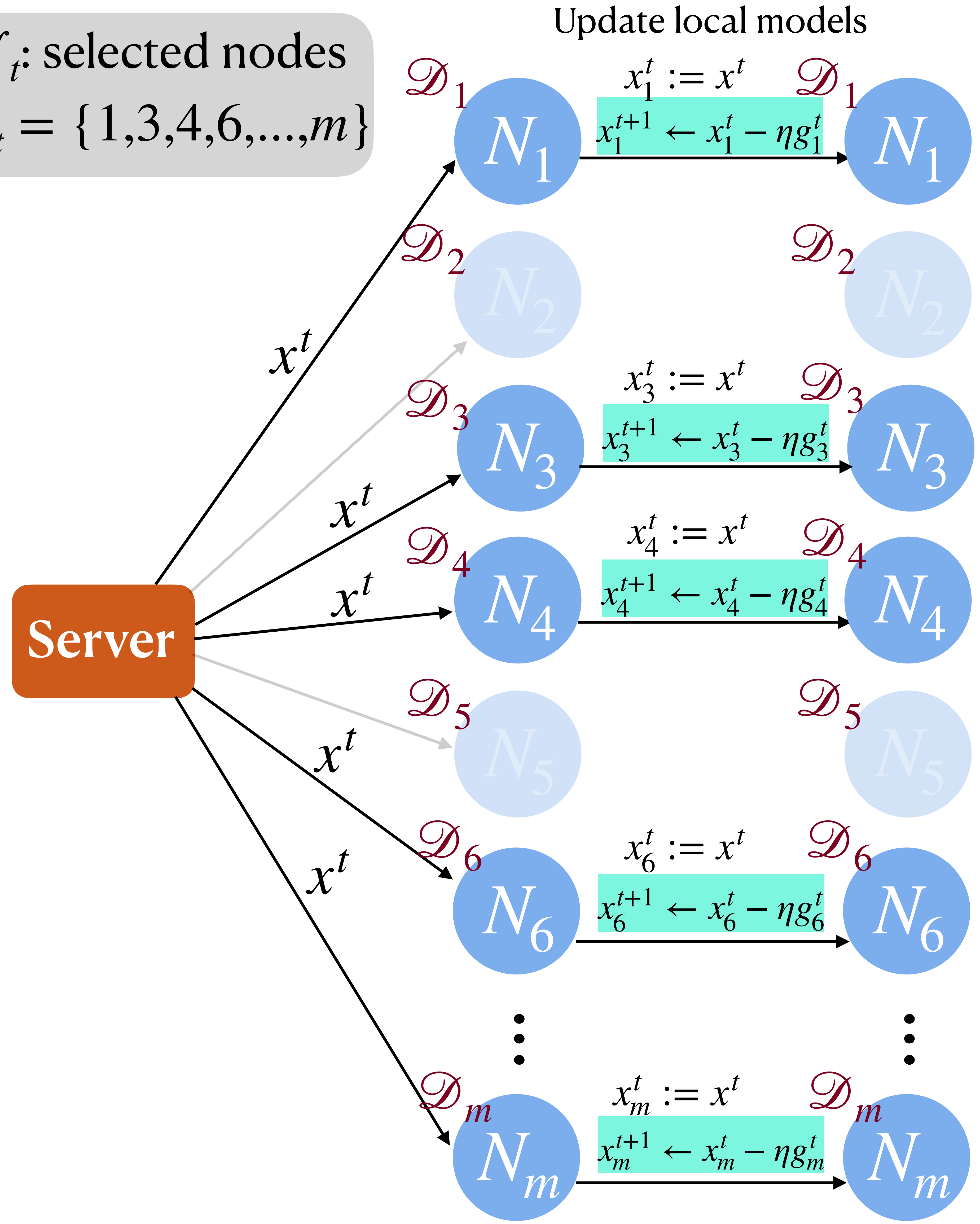


- $\mathcal{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

$$\arg\min_{x\in\mathbb{R}^d} \left(F(x) := \frac{1}{m}\sum_{i=1}^{m} F_i(x) \right)$$

Distributed Local SGD with Client Sampling

$\mathcal{K}_t$: selected nodes
$\mathcal{K}_t = \{1,3,4,6,\ldots,m\}$



- $\mathcal{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

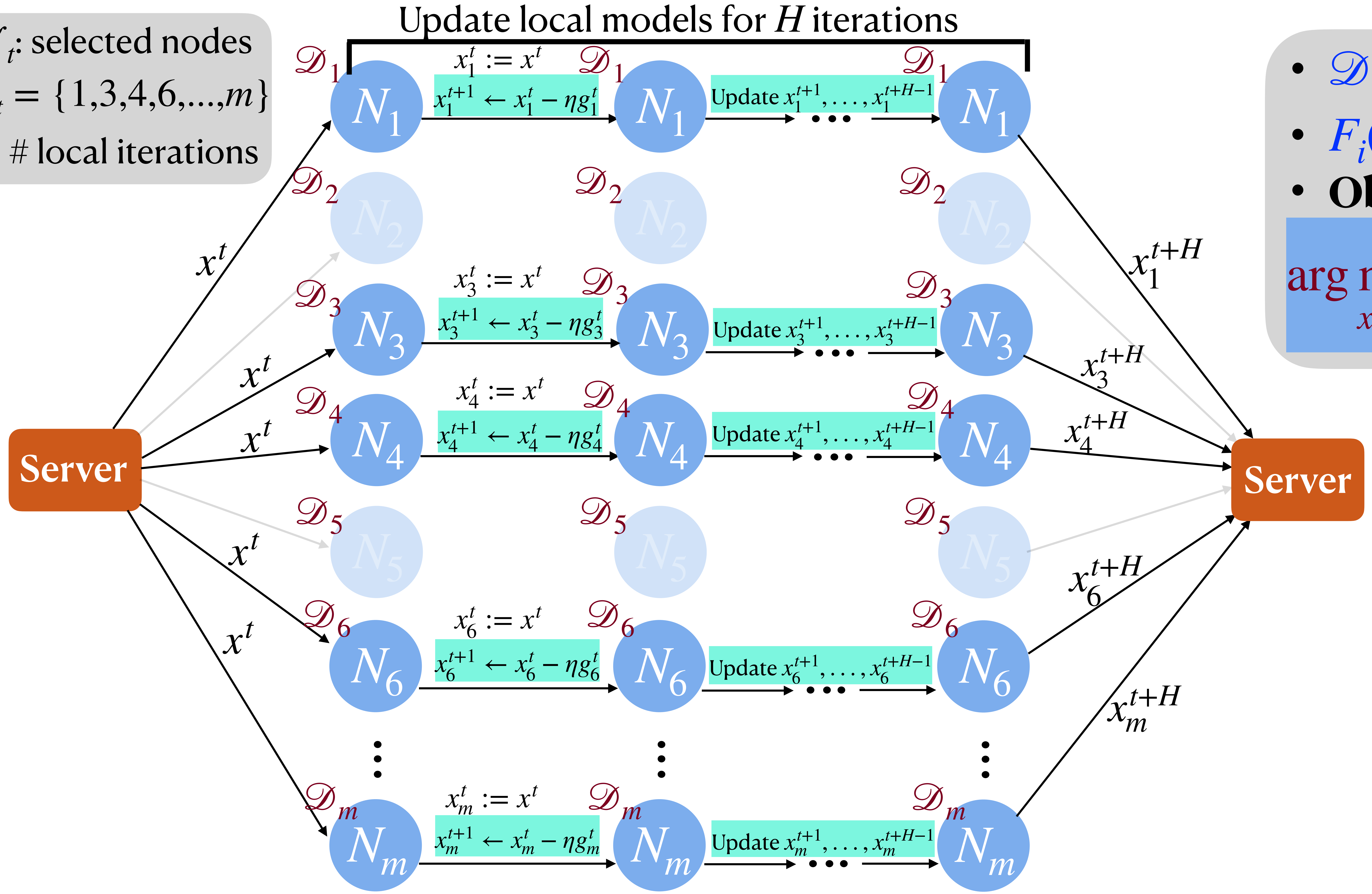
$$\arg\min_{x\in\mathbb{R}^d} \left(F(x) := \frac{1}{m}\sum_{i=1}^{m} F_i(x) \right)$$

Distributed Local SGD with Client Sampling

$\mathscr{K}_t$: selected nodes

$\mathscr{K}_t = \{1,3,4,6,\ldots,m\}$

H: # local iterations



- $\mathscr{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

$$\arg\min_{x \in \mathbb{R}^d} \left(F(x) := \frac{1}{m} \sum_{i=1}^{m} F_i(x) \right)$$

Distributed Local SGD with Client Sampling

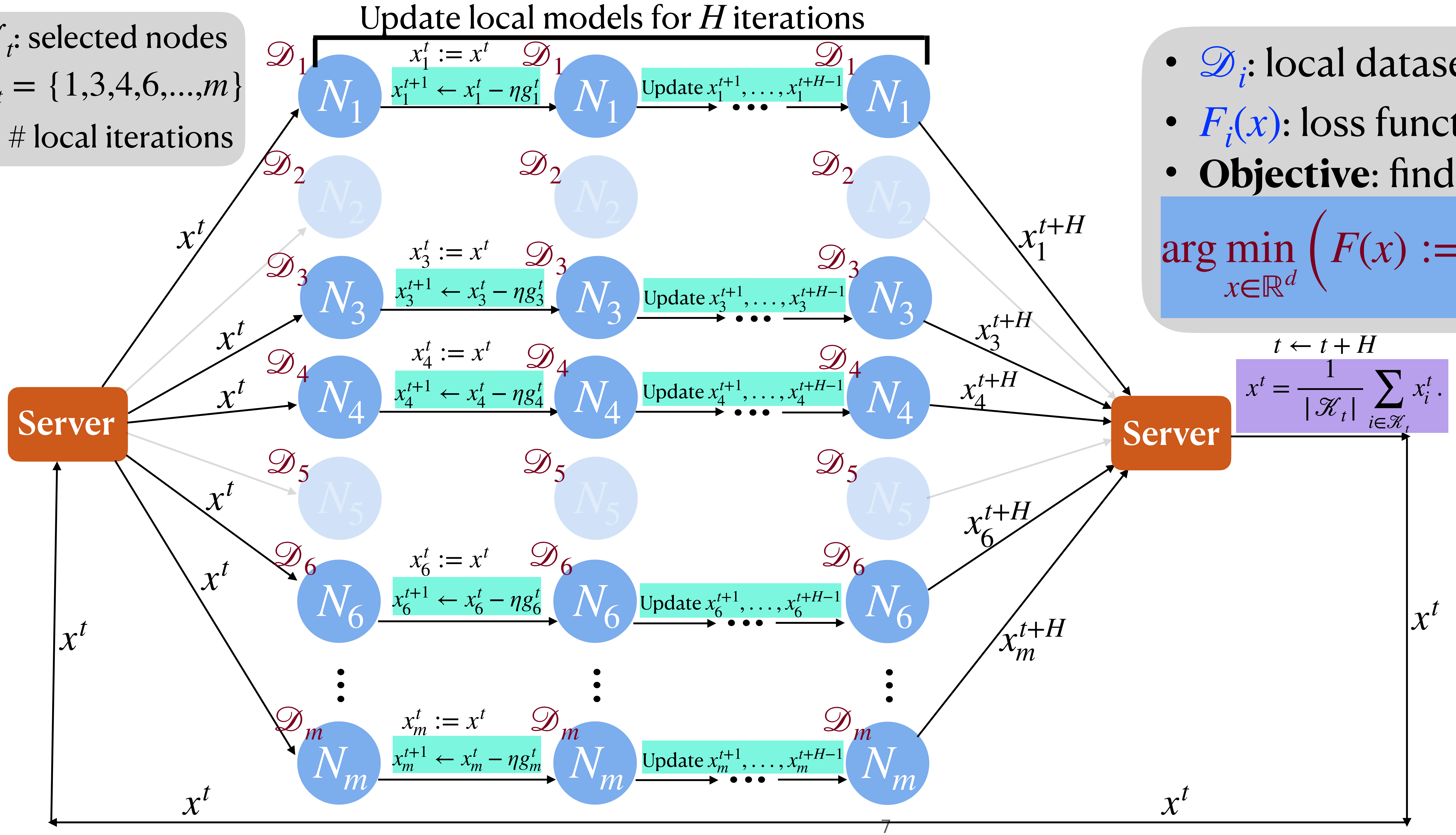
$\mathcal{K}_t$: selected nodes

$\mathcal{K}_t = \{1,3,4,6,\ldots,m\}$

H: # local iterations

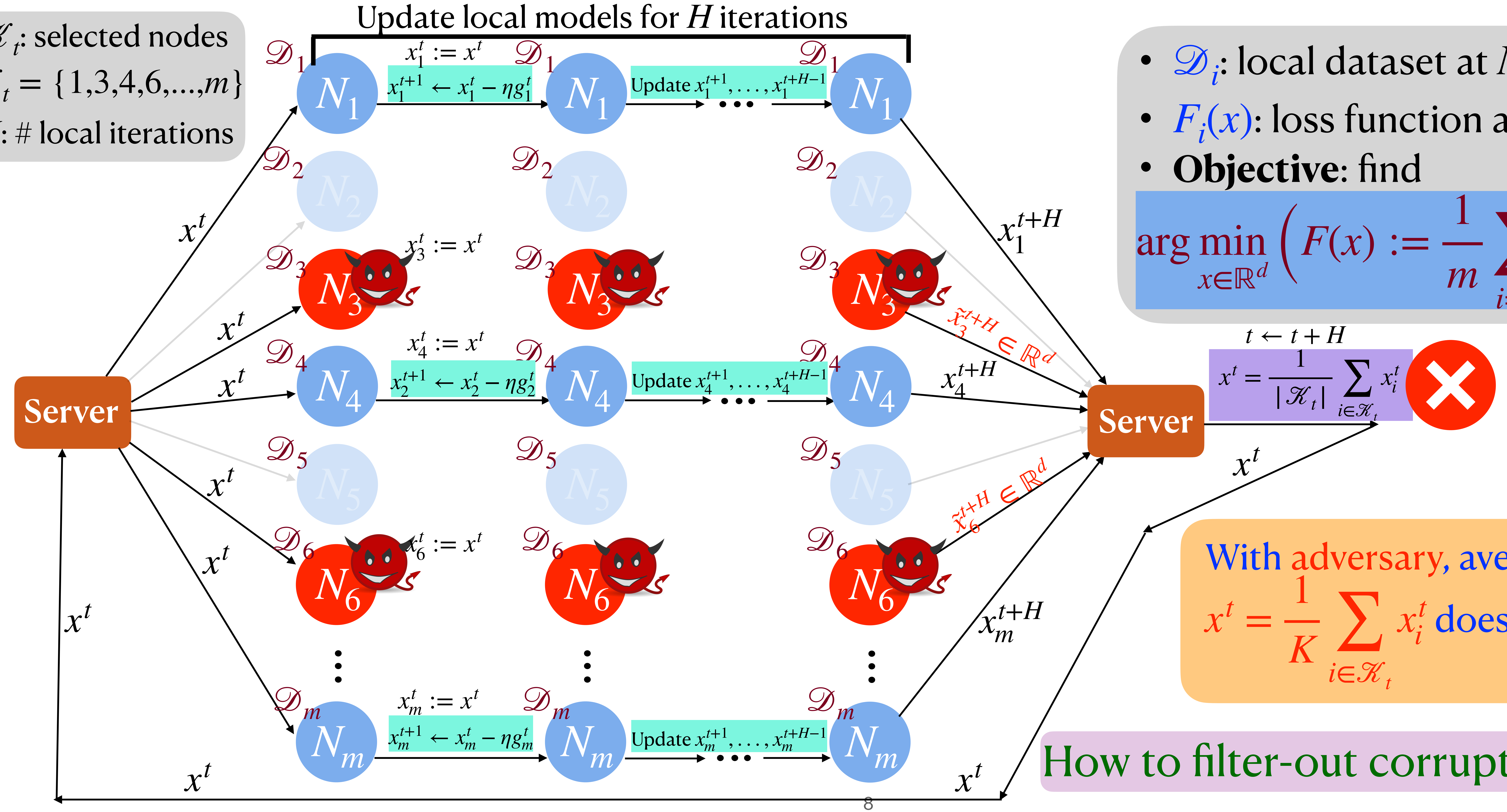
- $\mathcal{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

$$\arg\min_{x \in \mathbb{R}^d} \left(F(x) := \frac{1}{m} \sum_{i=1}^{m} F_i(x) \right)$$



Distributed Local SGD under Adversarial Attacks

$\mathcal{K}_t$: selected nodes
$\mathcal{K}_t = \{1,3,4,6,\ldots,m\}$
H: # local iterations



- $\mathcal{D}_i$: local dataset at N_i
- $F_i(x)$: loss function at N_i
- **Objective**: find

$$\arg\min_{x \in \mathbb{R}^d} \left(F(x) := \frac{1}{m} \sum_{i=1}^{m} F_i(x) \right)$$

With adversary, averaging $x^t = \frac{1}{K} \sum_{i \in \mathcal{K}_t} x_i^t$ does not work

How to filter-out corrupt vectors?

Existing Methods and Our Approach

Existing methods (not incorporating local iterations): Extensive literature

Norm-based filtering, Median or Trimmed-mean[Yin et al. ICML18,..], Coding-theoretic/redundancy-based solutions[Data et al. TIT21, Draco ICML18, Detox NeurIPS19,..], or other heuristics[Krum-17, Bulyan-18,...]

- Either give poor guarantees for high-dimensional model learning
- Or require strong unrealistic assumptions that are not feasible in federated learning

Existing methods (incorporating local iterations): Only one paper

Based on Trimmed-mean[SLSGD-19]: Poor guarantees, sub-optimality gap in optimization is Huge.

Our approach: Use high-dimensional robust mean estimation (RME) algorithm

- [Diakonikolas et al. FOCS 16, Lai et al. FOCS 16, Steinhardt et al. ITCS 18,...]
- Gives dimension-independent error guarantees for unit variance input vectors
- **Problem with RME algorithms:** Their analysis is only for i.i.d. data

Our contribution: Extend the analysis of RME algorithms from i.i.d. data to heterogeneous data and that work with stochastic gradients with local iterations

Main Technical Lemmas

L: smoothness parameter — H: # local iterations — σ^2: SGD variance — b: mini-batch size for SGD — ϵ: corruption threshold

κ^2: heterogeneity bound ($\|\nabla F_i(x) - \nabla F(x)\| \leq \kappa$) — d: model dimension — $t_1, \ldots, t_k$: synchronization indices — $K = |\mathcal{K}_t|$: #communicating nodes

Lemma (Bounding the Drift — for Local Iterations): If $\eta \leq 1/8HL$, then for any honest nodes $r \neq s$, we have

$$\sum_{t=t_k}^{t_{k+1}-1} \mathbb{E}\|x_r^t - x_s^t\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$$

Main Technical Lemmas

L: smoothness parameter H: # local iterations σ^2: SGD variance b: mini-batch size for SGD ϵ: corruption threshold

κ^2: heterogeneity bound ($\|\nabla F_i(x) - \nabla F(x)\| \leq \kappa$) d: model dimension $t_1, \ldots, t_k$: synchronization indices $K = |\mathcal{K}_t|$: #communicating nodes

Lemma (Bounding the Drift — for Local Iterations): If $\eta \leq 1/8HL$, then for any honest nodes $r \neq s$, we have $\sum_{t=t_k}^{t_{k+1}-1} \mathbb{E}\|x_r^t - x_s^t\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$

Lemma (Robust Parameter Estimation — for Combating Byzantine Updates):

Matrix concentration: W.h.p., $\exists$ a subset $\mathcal{S} \subset \mathcal{K}_{t_k}$ of honest nodes of size $(1-\epsilon)K \geq \frac{2K}{3}$, s.t.

$$\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(x_i^{t_k} - x_{\mathcal{S}}^{t_k})(x_i^{t_k} - x_{\mathcal{S}}^{t_k})^T\right) \leq \sigma_0^2 := O\left(\frac{H^2\sigma^2 d}{bK} + H^2\kappa^2\right), \text{ where } x_{\mathcal{S}}^{t_k} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} x_i^{t_k}$$

Outlier-filtering algorithm[Diakonikolas et al. FOCS16, Lai et al. FOCS16, Steinhardt et al. ITCS18,...]: We can find an estimate $\widehat{x}$ of $x_{\mathcal{S}}^{t_k}$ in polynomial time s.t. $\|\widehat{x} - x_{\mathcal{S}}^{t_k}\| \leq O(\sigma_0\sqrt{\epsilon})$

Convergence Results

Theorem (Convergence Results)

Let $\epsilon < 1/3$ and $\eta = 1/8HL$. With prob. $(1 - (T/H)e^{-cK})$ for some const. $c > 0$, we have

- If F is L-smooth and μ-strongly convex:

$$\mathbb{E}\|x_T - x^*\|_2^2 \leq \left(1 - \frac{\mu}{16HL}\right)^T \|x_0 - x^*\|_2^2 + O\left(\frac{H\sigma^2 d}{bK} + H\kappa^2\right)$$

- If F is L-smooth (and non-convex):

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\|\nabla F(x_t)\|_2^2 \leq \frac{8HL^2}{T}\|x_0 - x^*\|_2^2 + O\left(\frac{H\sigma^2 d}{bK} + H\kappa^2\right)$$

- Approximation error consists of two terms (both have only linear dependence on H):

(i) $O\left(\frac{H\sigma^2 d}{bK}\right)$ — due to adversary and SGD (ii) $O(H\kappa^2)$ — due to data heterogeneity

Numerical Results

- Training of one-layer neural network on MNIST dataset with $H = 7$ local iterations
- Heterogeneous data distributions
- Compared with coordinate-wise median/trimmed-mean, Krum, Bulyan, NoAttack_NoDecoding

- Our algorithm (in red) beats other methods (see the paper for more attacks and details)

Thank you for your attention!