Federated Learning: Methodologies, Challenges and Opportunities

Anastasia Koloskova

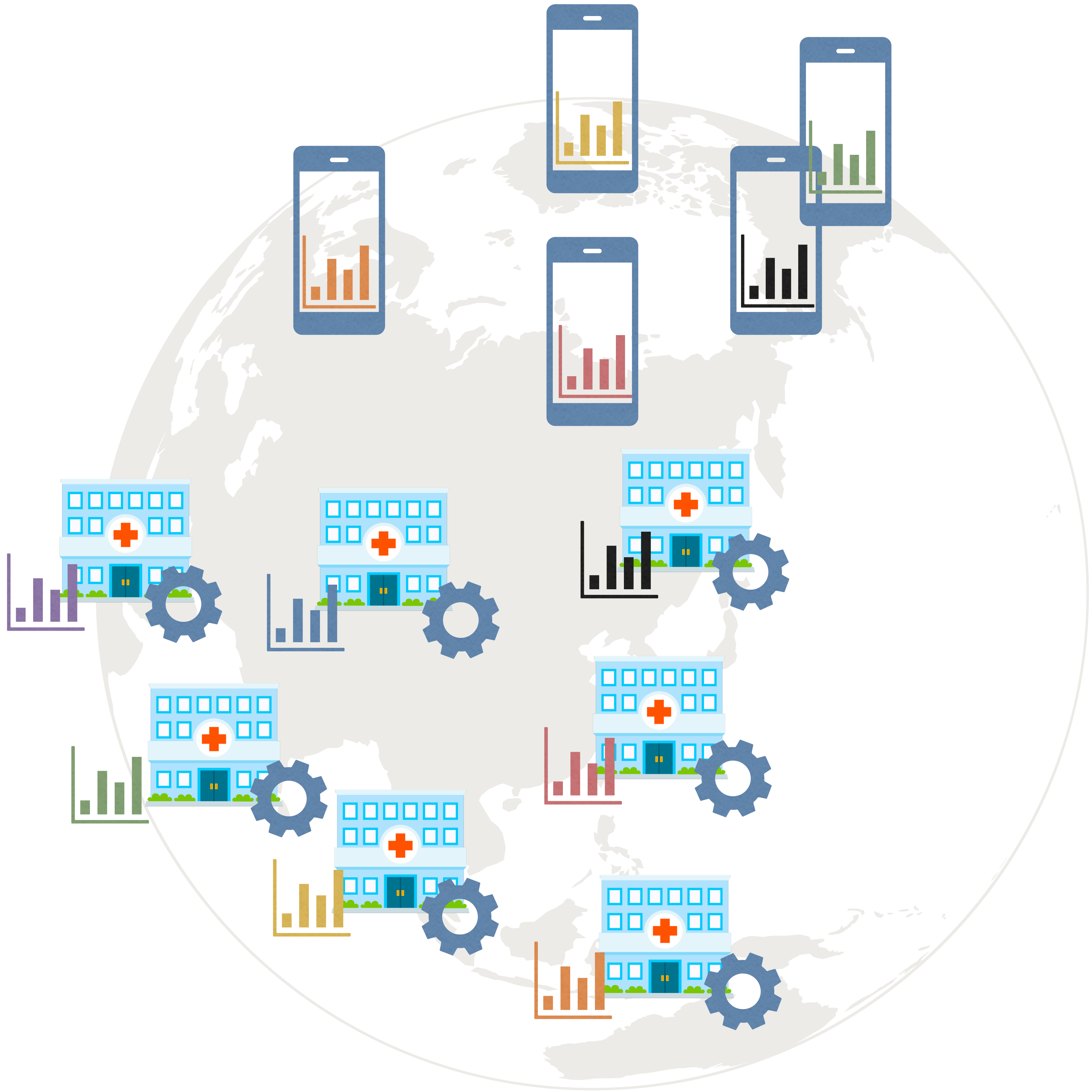
6 November

Motivation

- In many applications data originally comes from distributed sources
- Two examples:
  - Text generated on people's smartphones
  - Medical data (e.g. imaging) collected at different hospitals

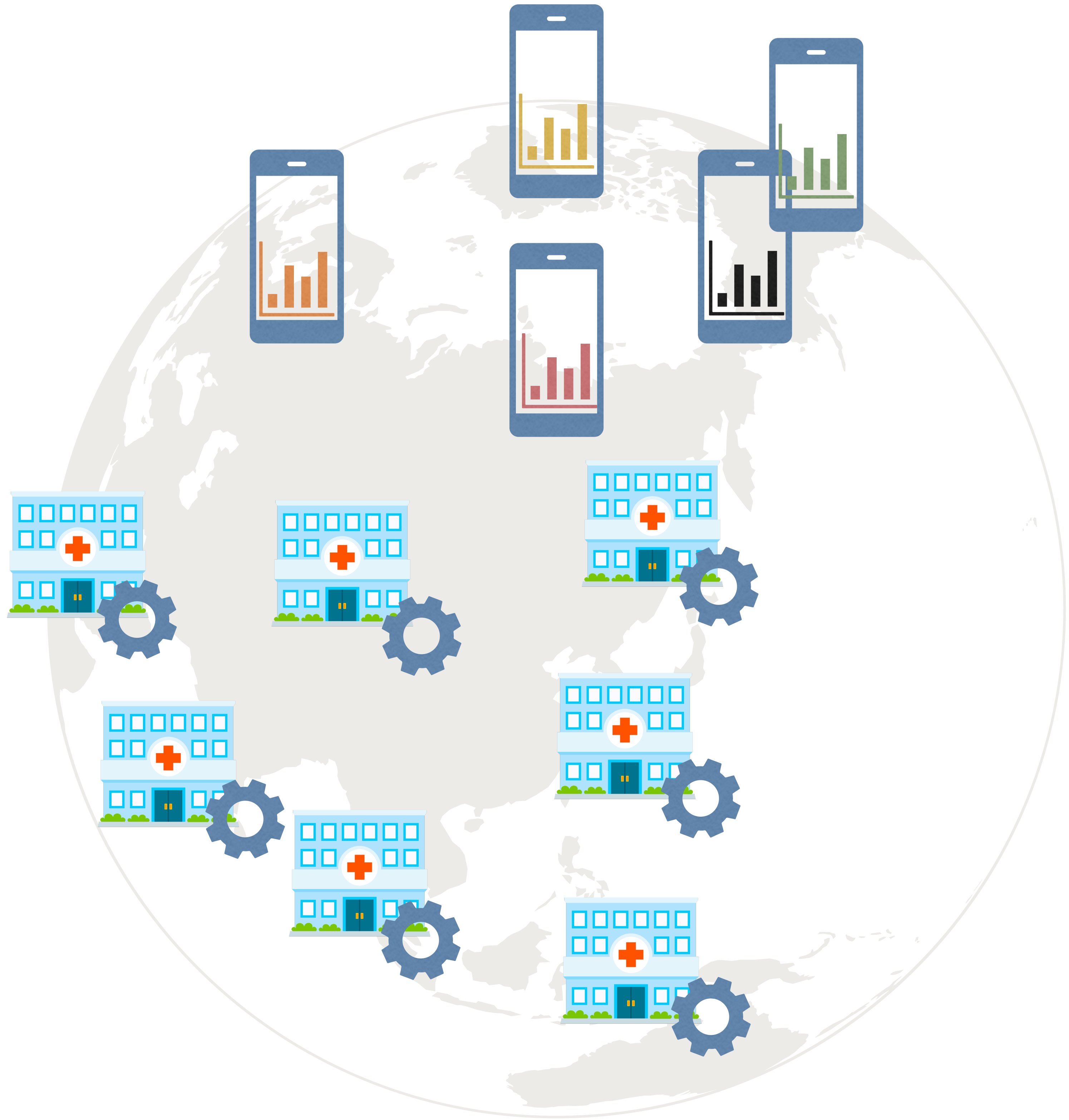
Classical Approach

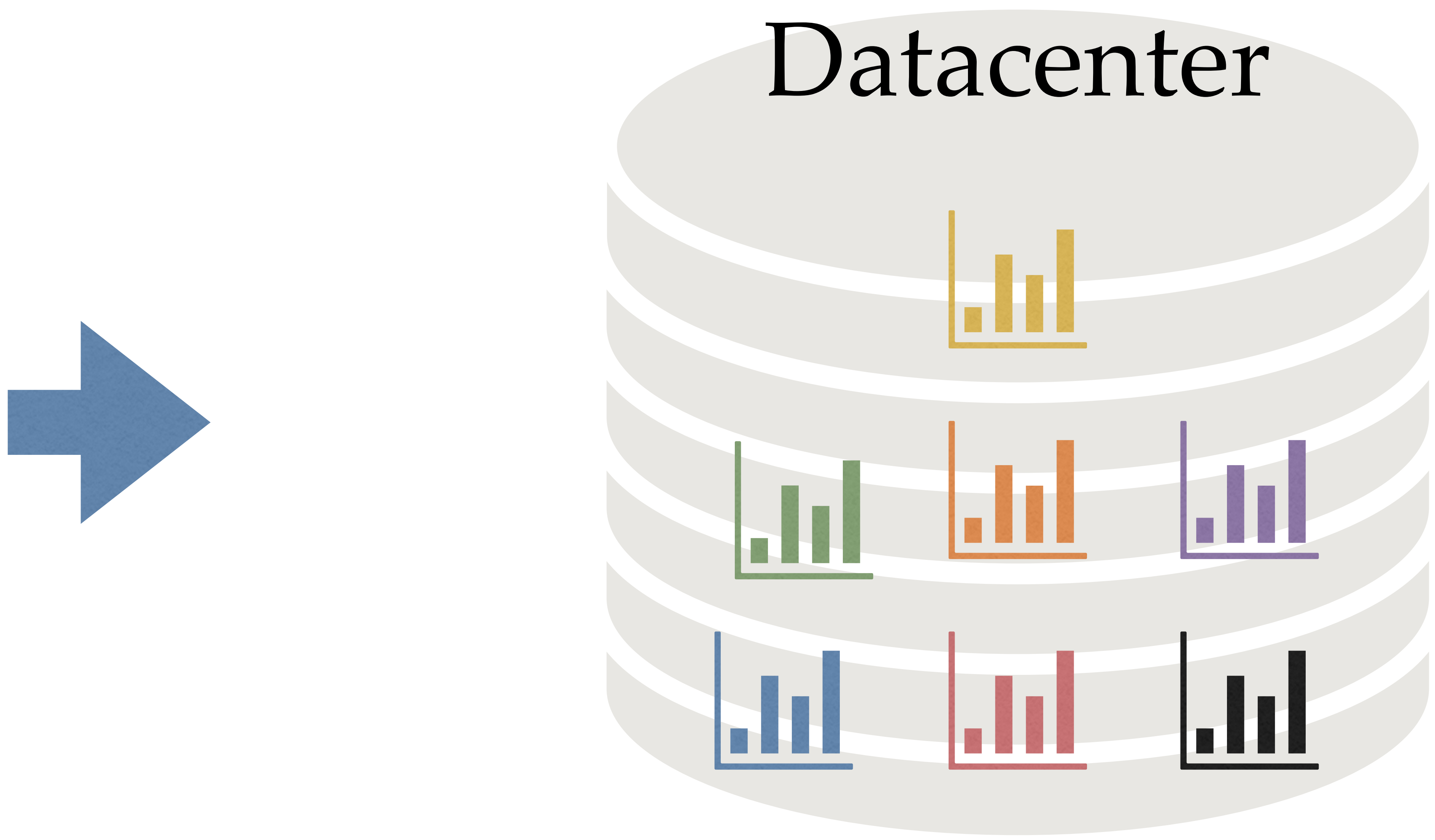
Collect all the training data in a datacenter



Classical Approach

Collect all the training data in a datacenter





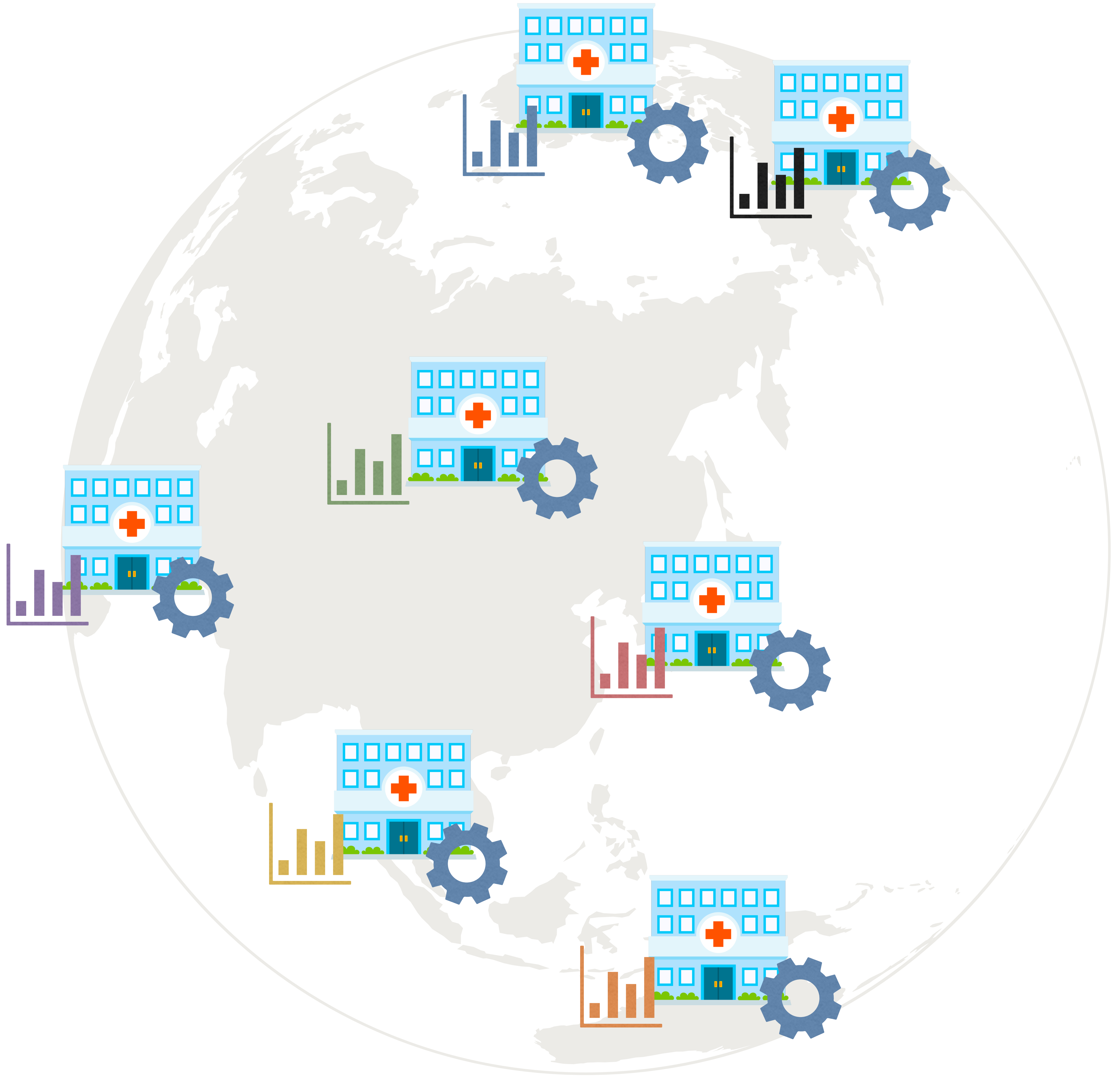
Classical Approach

Collect all the training data in a datacenter

Might not be possible or desirable due to privacy constraints

Federated Learning

Collaboratively learn from the data directly on devices/organizations without communicating raw training data outside



Mathematical Formulation

$\mathscr{D}_i$ f_i

Local data

Local loss based on local data

Goal: to collaboratively solve a common ML task based on private local data

Mathematical Formulation

$$f_i(x) = \frac{1}{|D_i|} \sum_{\xi \in D_i} loss(x, \xi)$$

parameters of ML model   local datapoints

$\mathscr{D}_i$ f_i

Local data   Local loss based on local data

Goal: to collaboratively solve a common ML task based on private local data

Mathematical Formulation

Distributed objective function: $$\min_{\boldsymbol{x}} \ f(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} f_i(\boldsymbol{x})$$

ML model we want to learn

Number of clients

$\mathscr{D}_i$ f_i

Local data

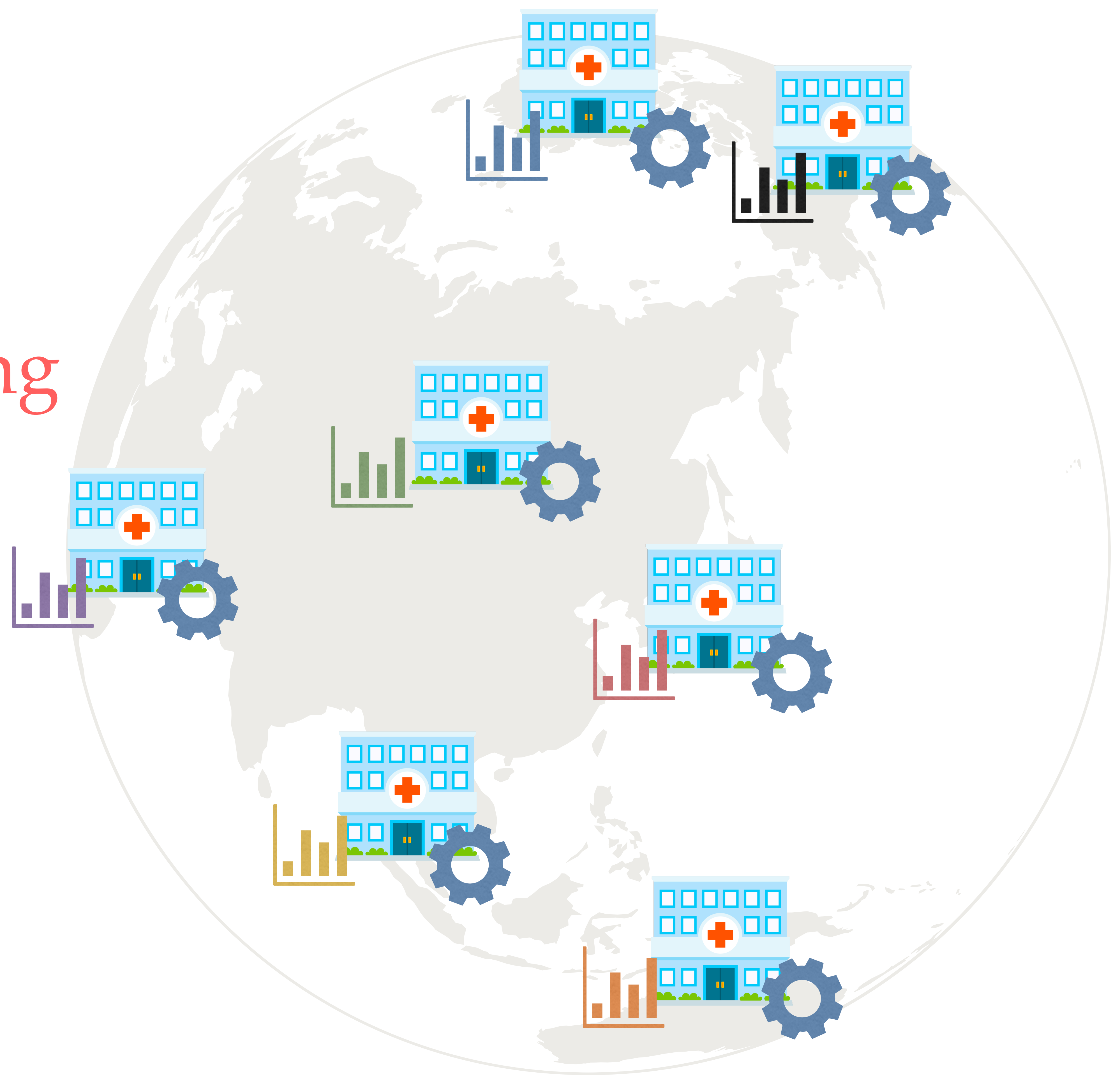
Local loss based on local data

Goal: to collaboratively solve a common ML task based on private local data

Federated Learning

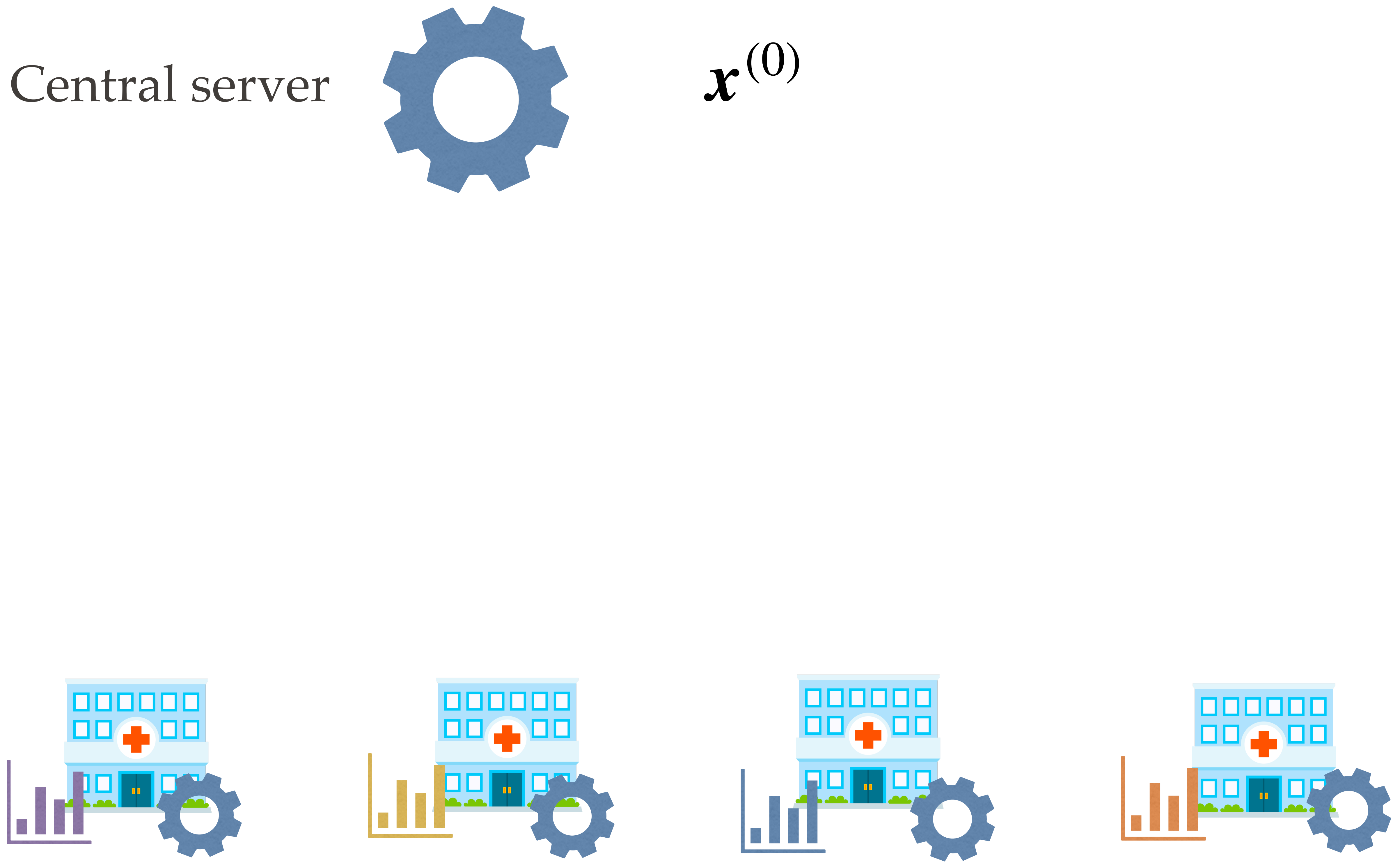
The most popular algorithm: Federated Averaging

(McMahan et al, 2017)
(Konecncy et al, 2016)



Learning Procedure: Federated Averaging

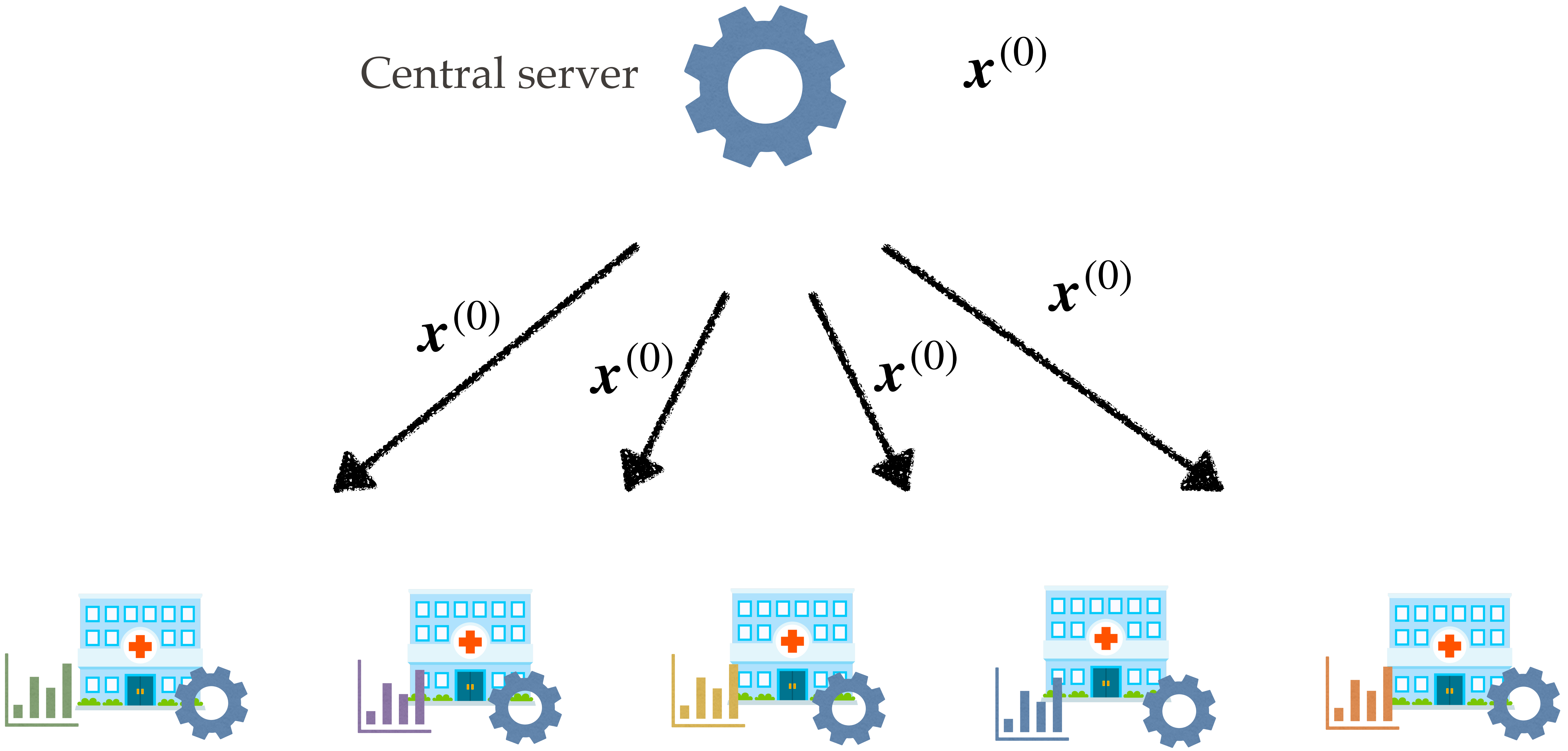
(McMahan et al, 2017)



Sever choses the model architecture, and initialises it

Learning Procedure: Federated Averaging

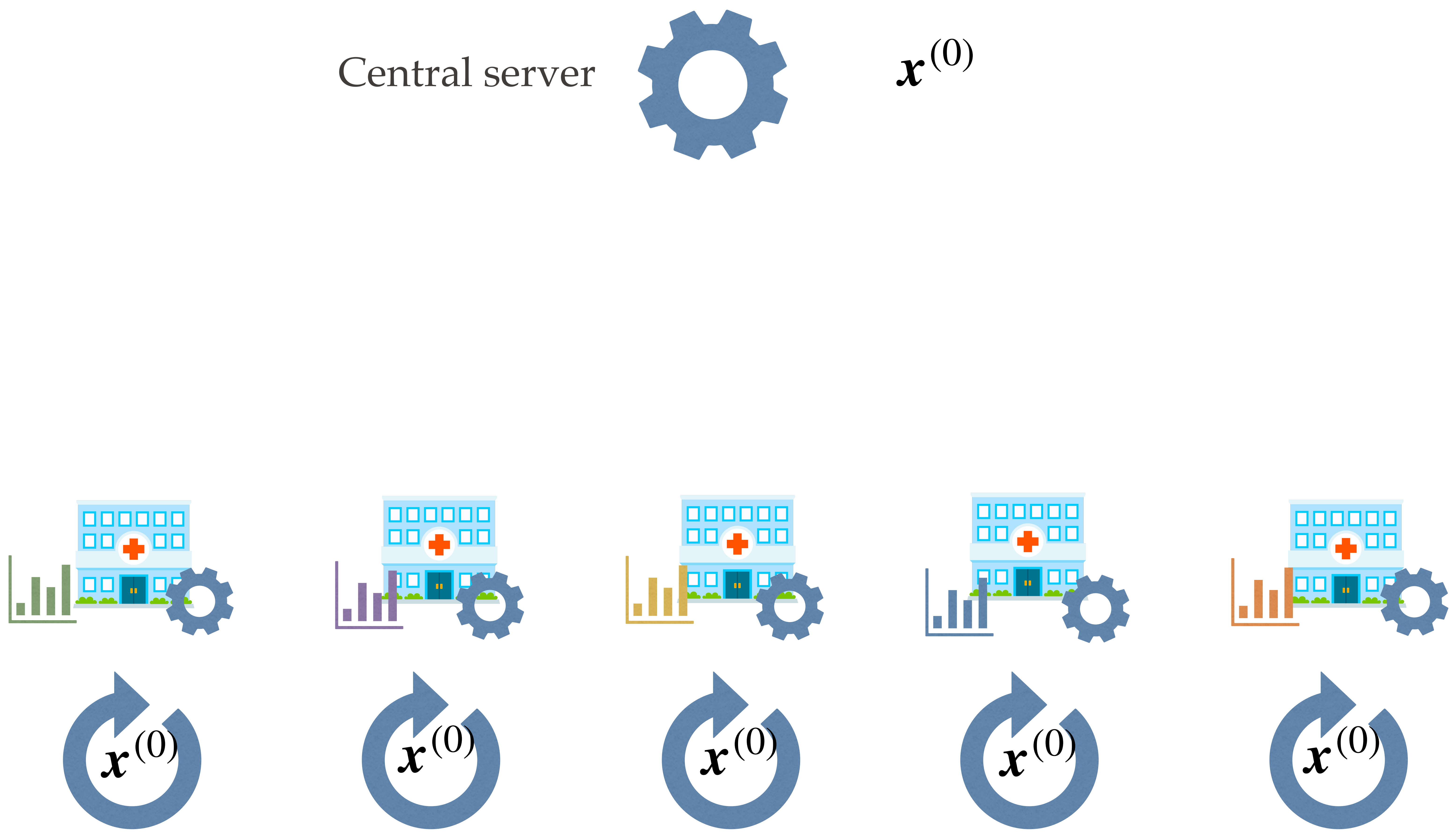
(McMahan et al, 2017)



Sends this model to all the participants

Learning Procedure: Federated Averaging

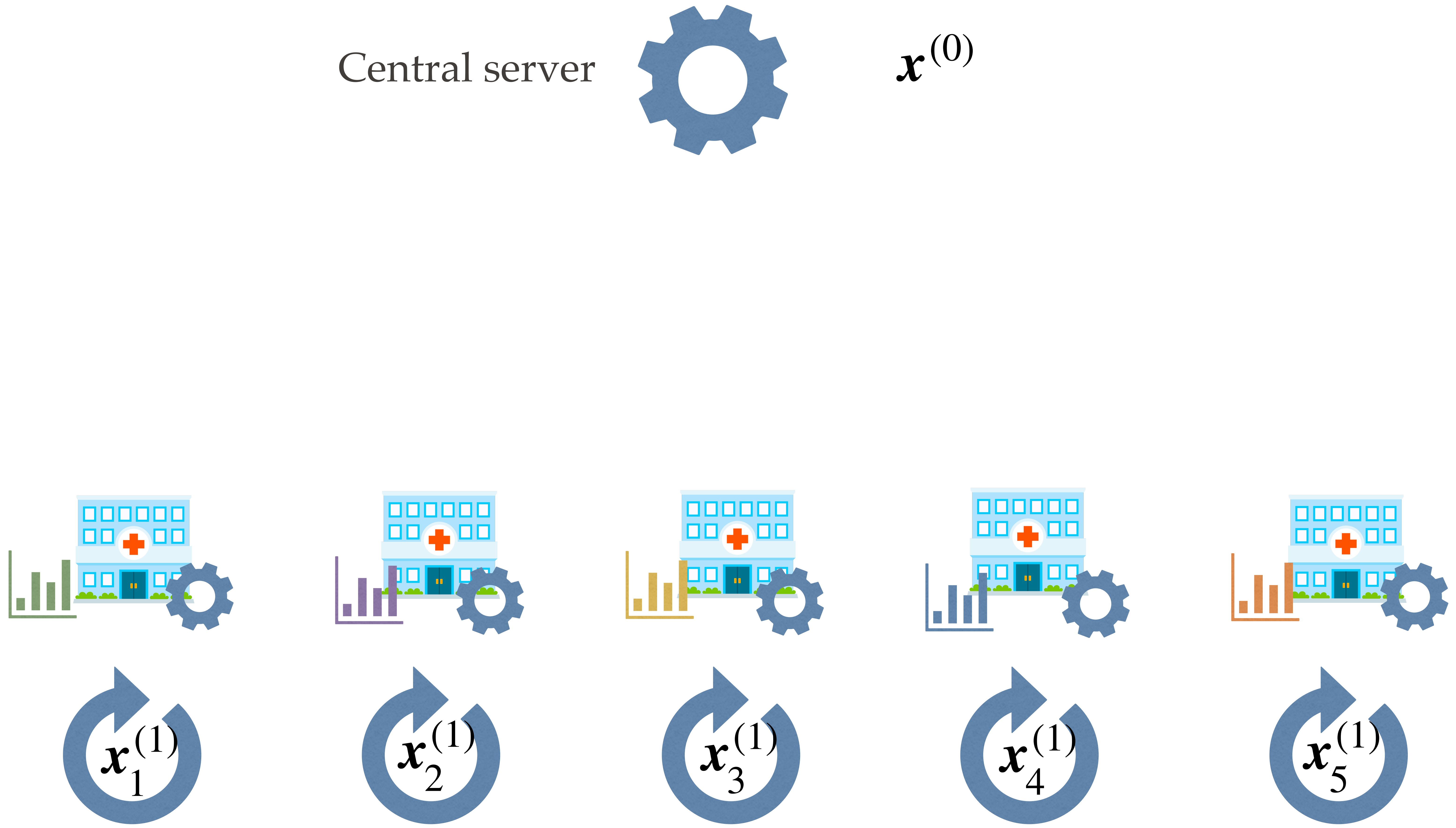
(McMahan et al, 2017)



Clients are performing local update steps based on the local data

Learning Procedure: Federated Averaging

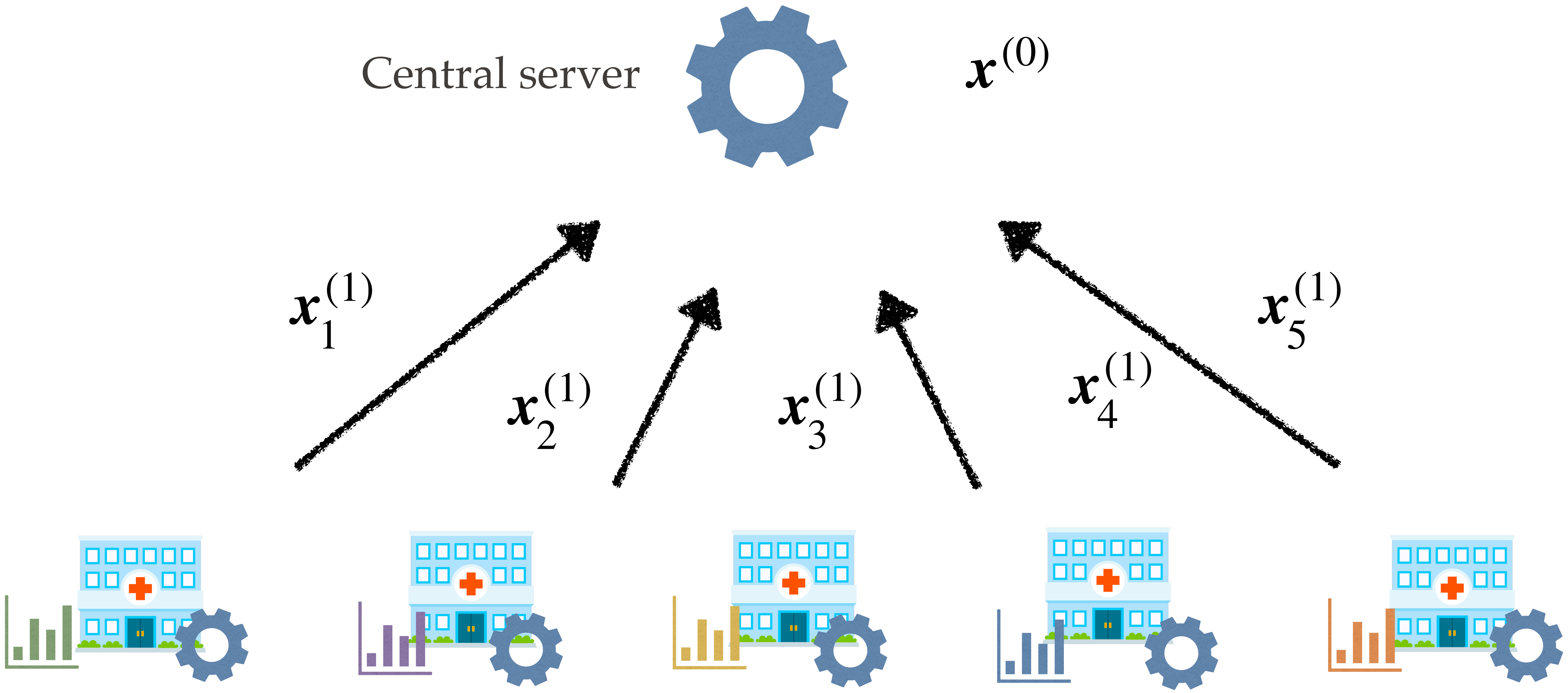
(McMahan et al, 2017)



Clients are performing local update steps based on the local data

Learning Procedure: Federated Averaging

(McMahan et al, 2017)

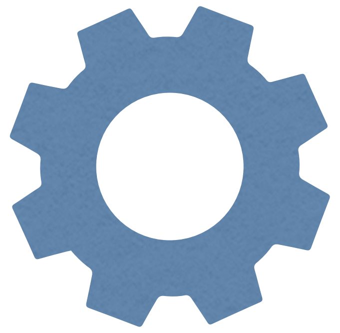


Send updated models to the server

Learning Procedure: Federated Averaging

(McMahan et al, 2017)

Central server

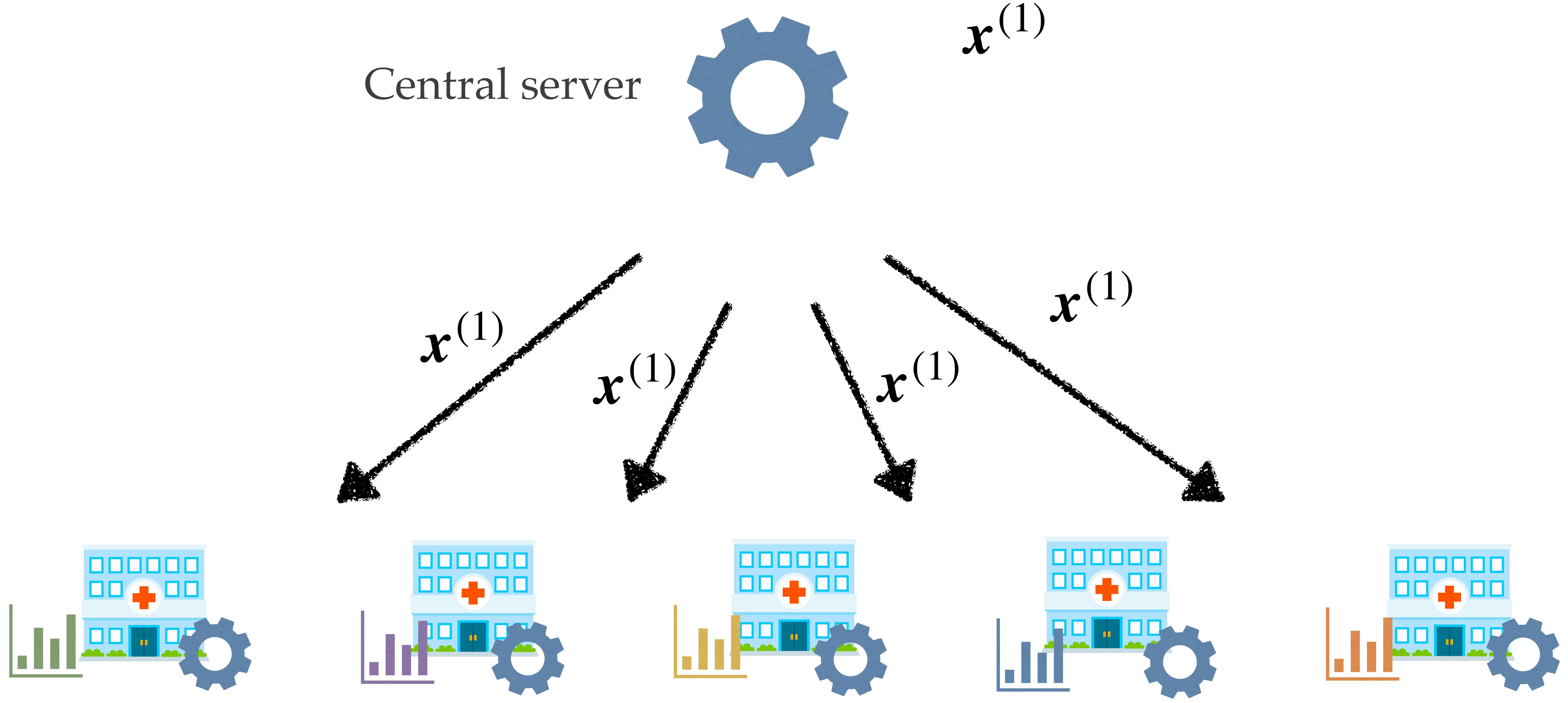


$$\boldsymbol{x}^{(1)} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i^{(1)}$$

Server averages the updates & updates the global model

Learning Procedure: Federated Averaging

(McMahan et al, 2017)



Procedure continues for many rounds

Challenges with Federated Averaging

Data heterogeneity

Local data are different

Communication is slow

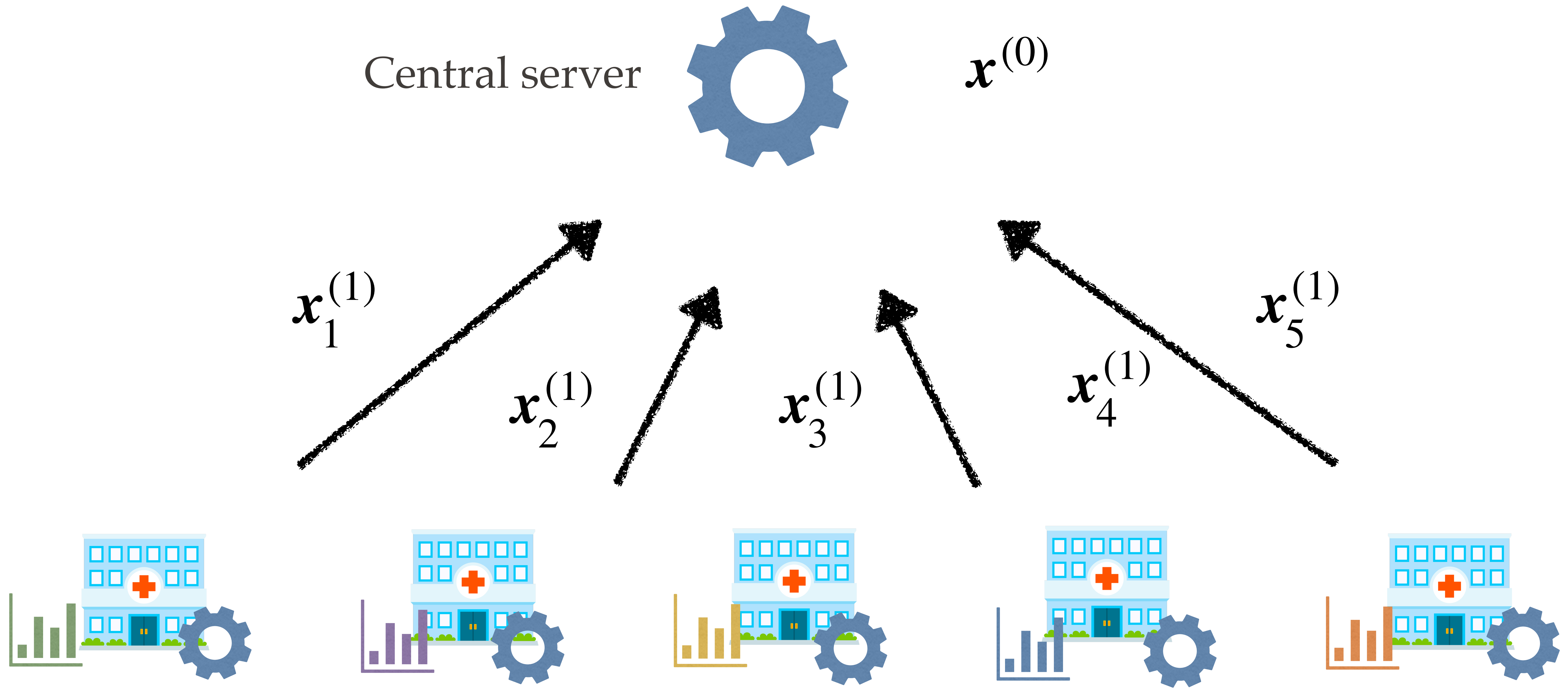
Need to do a lot of rounds
Hundreds of MB per model

Privacy

Frequently local data are sensitive & protected by privacy laws

(Kairouz et al, 2019)

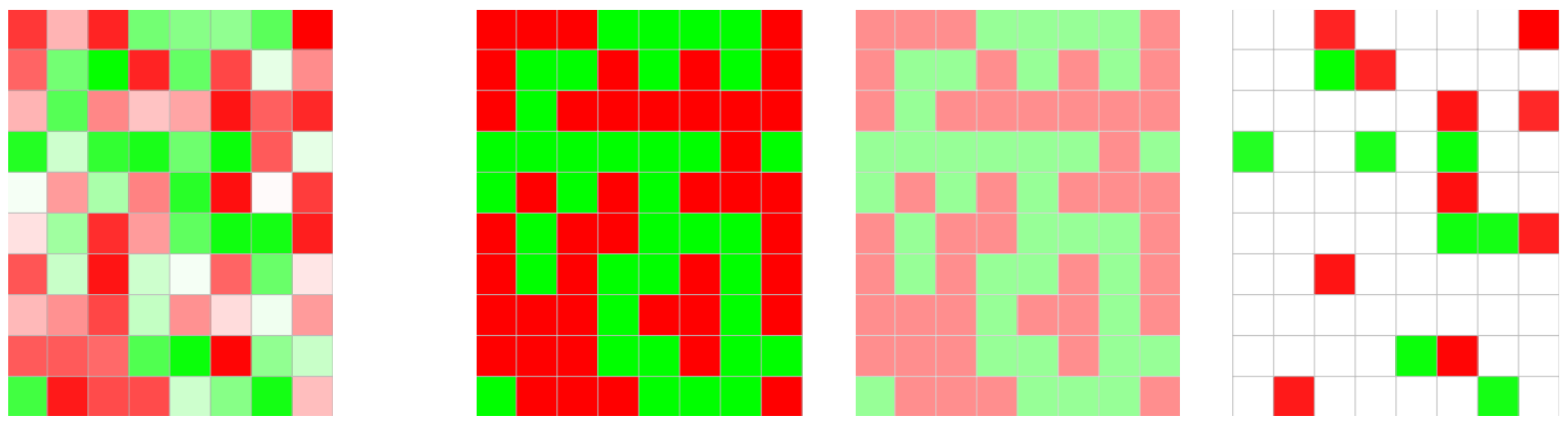
Communication is Slow



Solution 1: Communication Compression

$\boldsymbol{x}_i^{(t)}$ original model

$\text{Compressed}\left(\boldsymbol{x}_i^{(t)}\right)$



sign   sign + norm   top-k

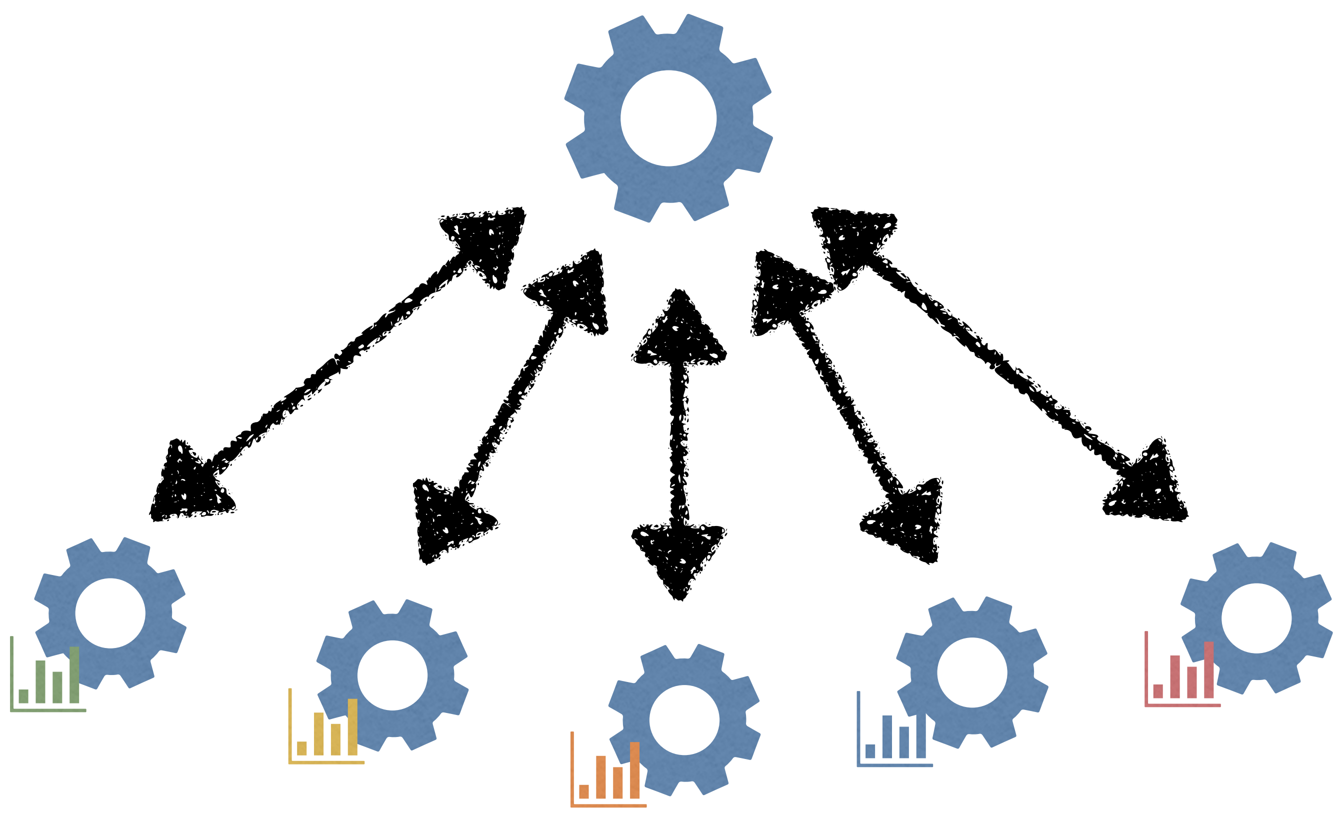
(Alistarh et al, 2017)

(Stich et al, 2018)

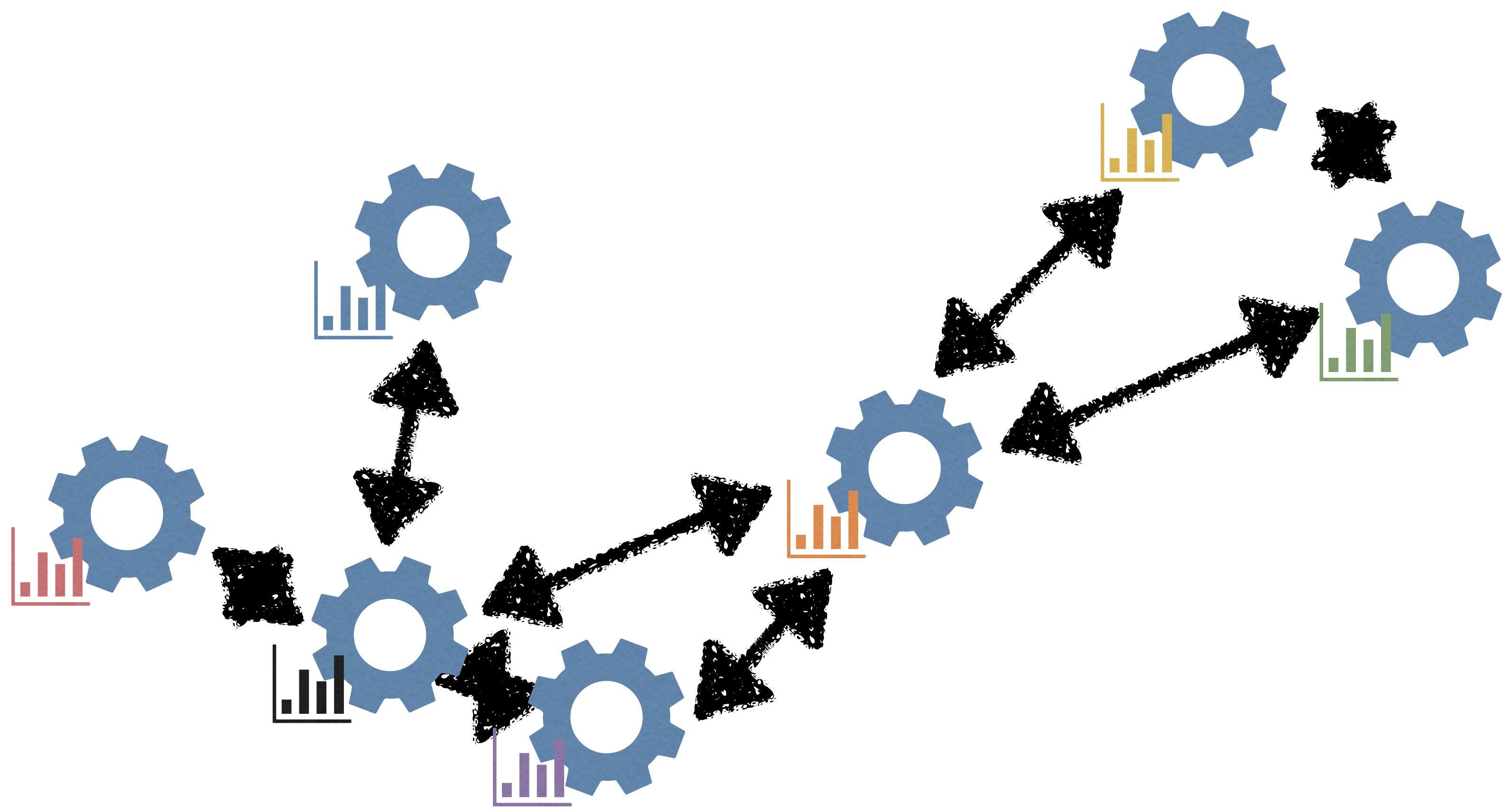
Need to make sure that optimisation is not hurt

Solution 2: Decentralized Communications

Centralized



Decentralized

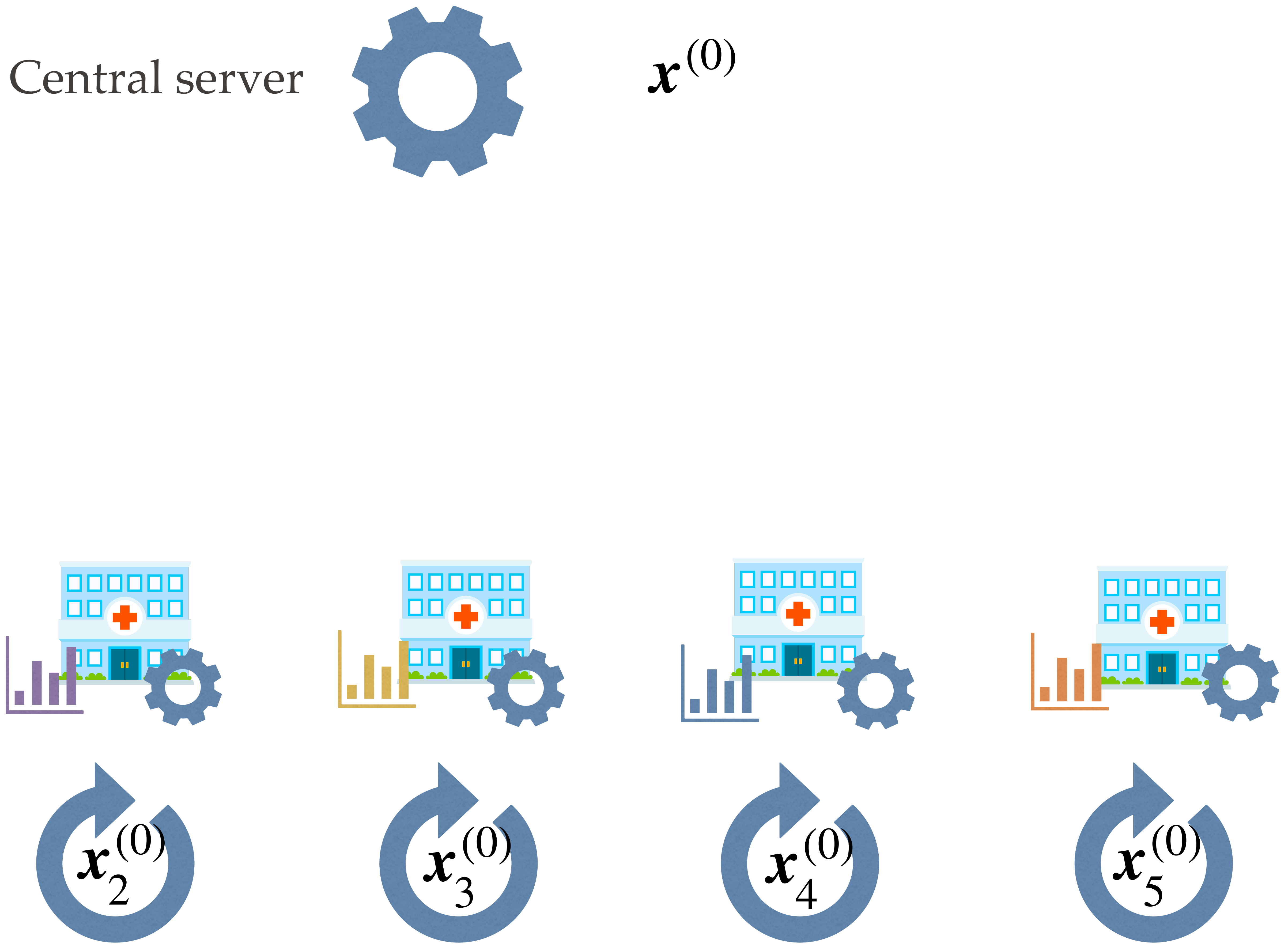


(Lian et al, 2017)

If the graph is sparse, improves communication time

Solution 3: Local Update Steps

(McMahan et al, 2017)



Perform many local update steps before communicating

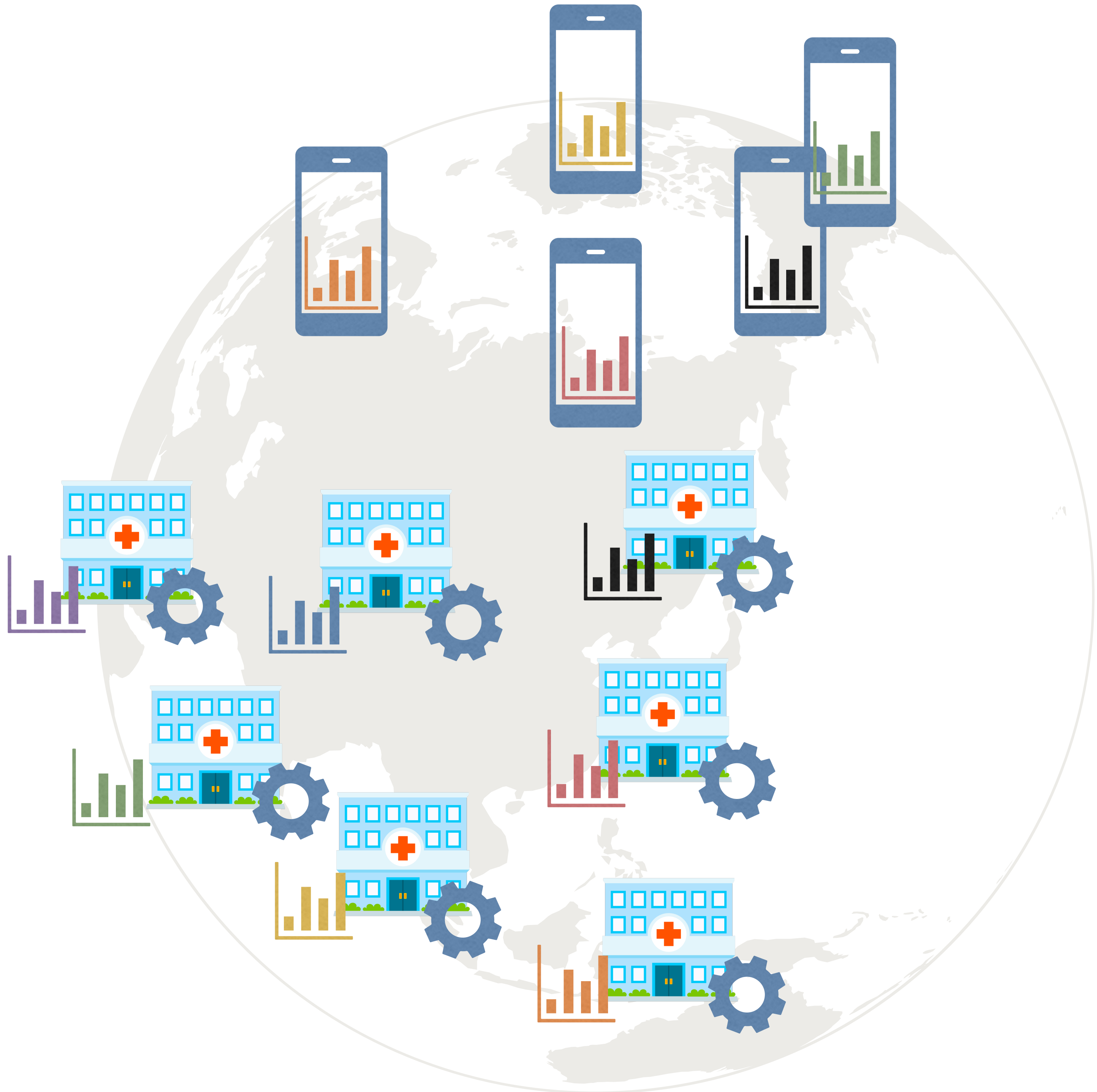
Challenges in Federated Learning

Communication is slow

Need to do a lot of rounds
Hundreds of MB per model

Data heterogeniety

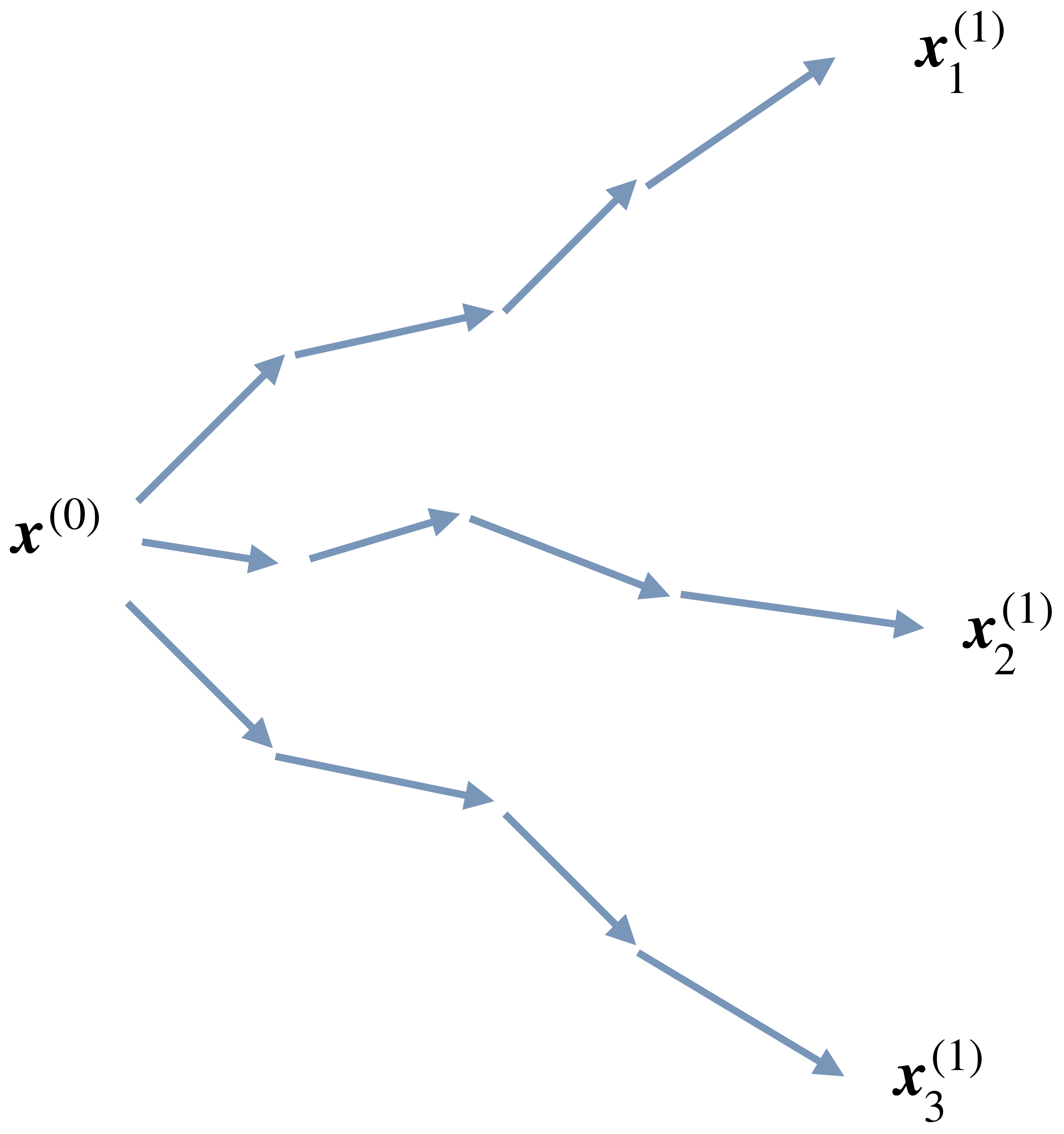
Local data are different

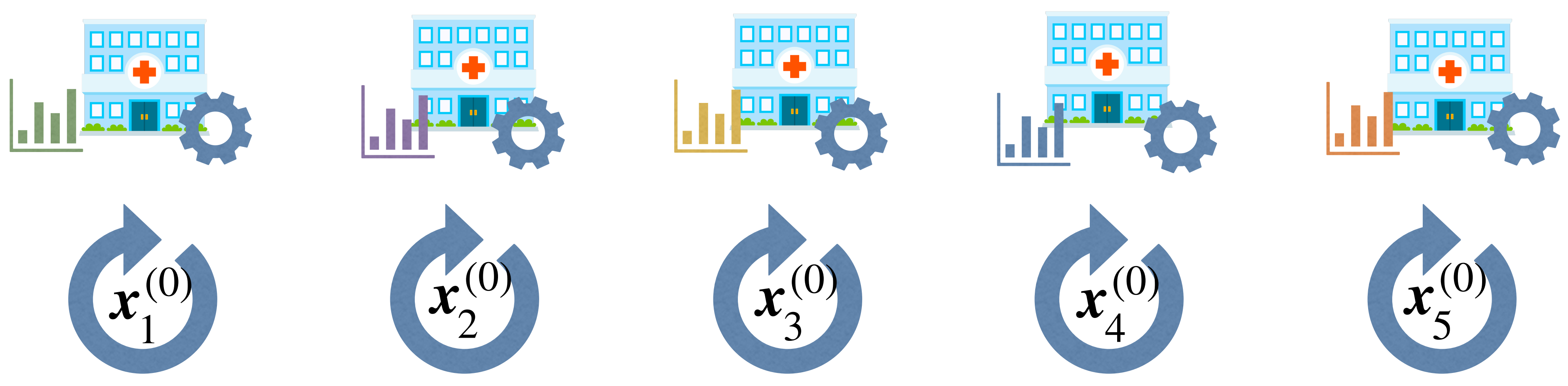


Privacy

Frequently local data are sensitive & protected by privacy laws

Data Heterogeneity

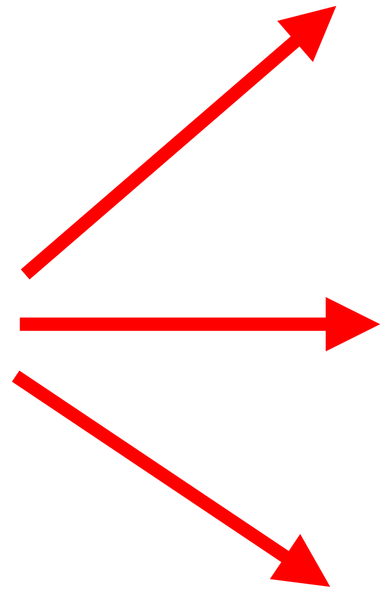




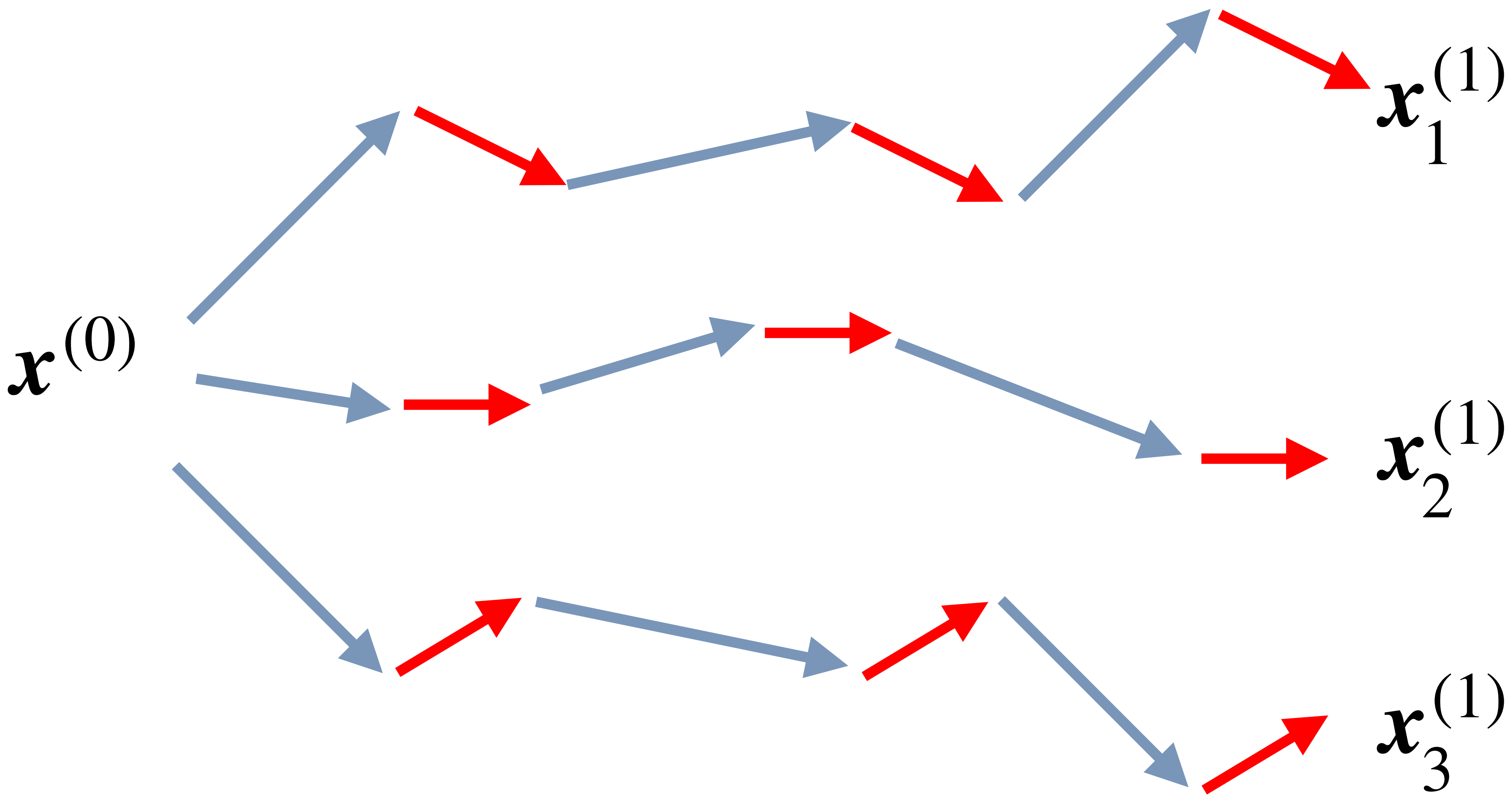
(Li et al, 2018)

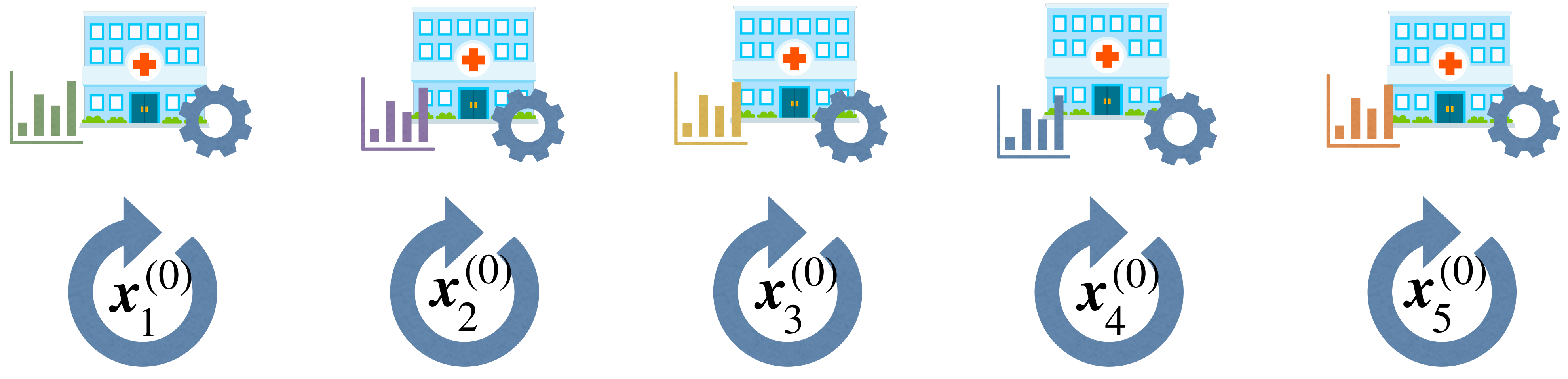
During the local steps models drift apart to fit the local data

Solution 1: Correct for the Drift



Estimated local drifts



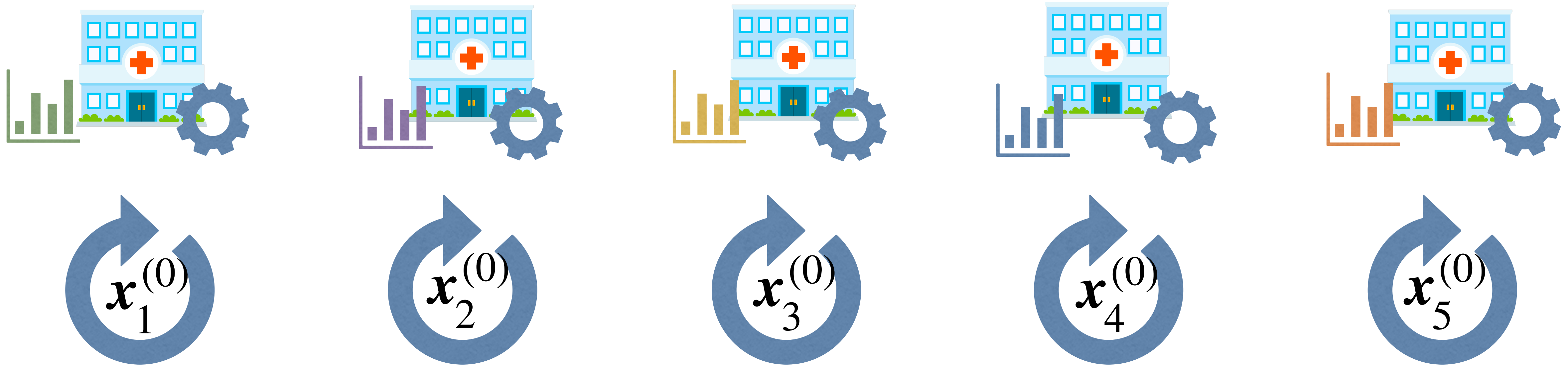


(Karimireddy et al, 2020)
(Li et al, 2019)

Estimate the local drift, and counter-balance it

Solution 2: Personalised Models

Not one global model, but learn many client-specific models



(Fallah et al, 2020)
(Chen et al, 2019)

How to efficiently use the data of the other participants

Challenges in Federated Learning

Data heterogeneity

Local data are different

Communication is slow

Need to do a lot of rounds
Hundreds of MB per model

Privacy

Frequently local data are sensitive & protected by privacy laws

Privacy in Federated Learning

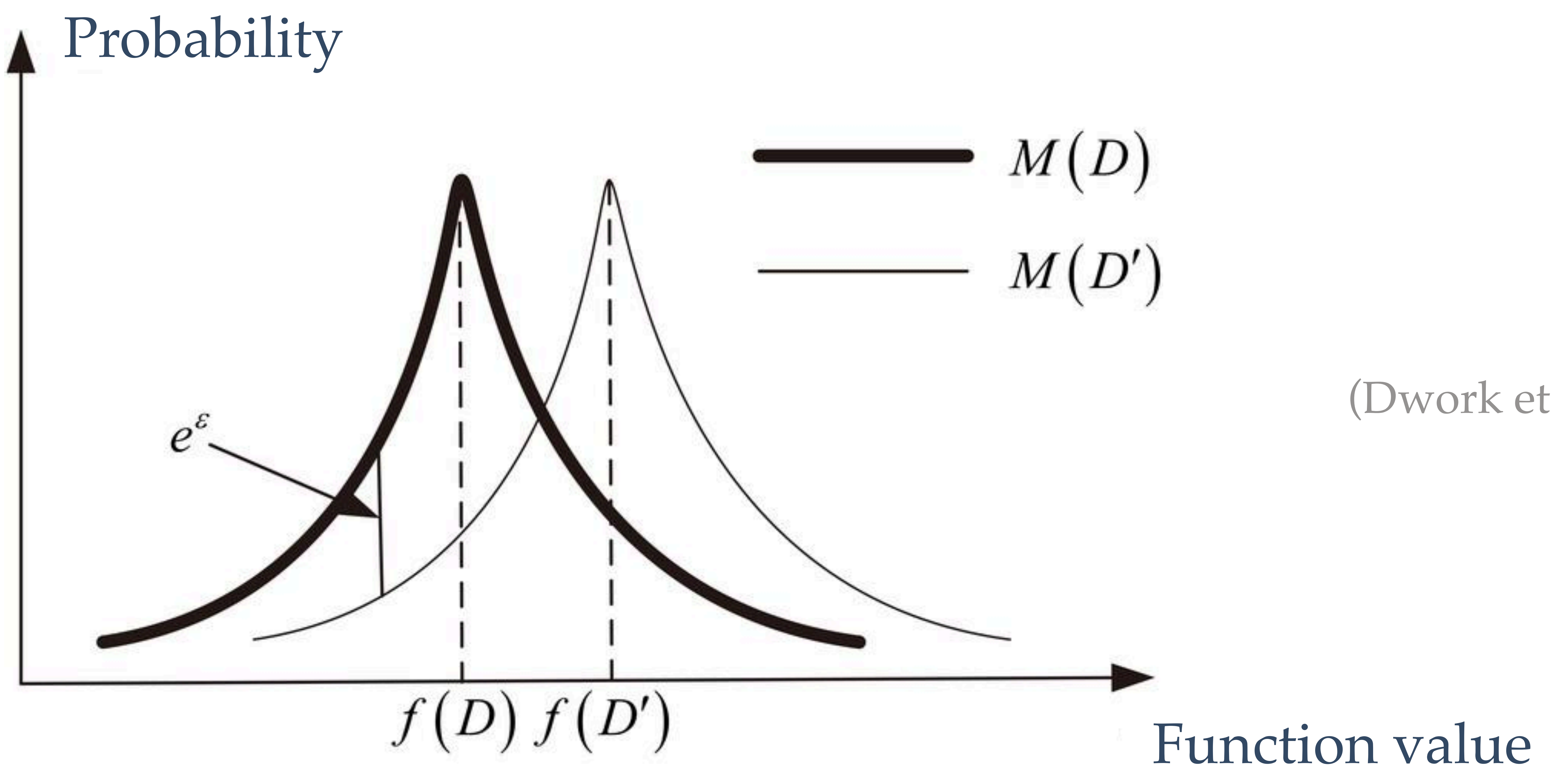
Frequently local data are sensitive & protected by privacy laws

Models and model updates might leak some information about the data

Differential Privacy

Formal definition of privacy

D and D′ are the two datasets that differ only in one datapoint

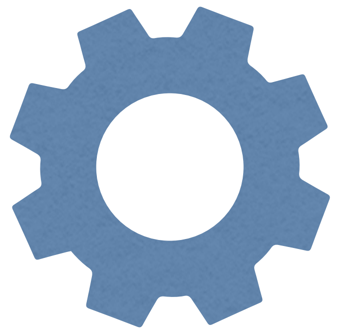


(Dwork et al, 2006)

Output distributions are ε-close

FedAvg with Differential Privacy

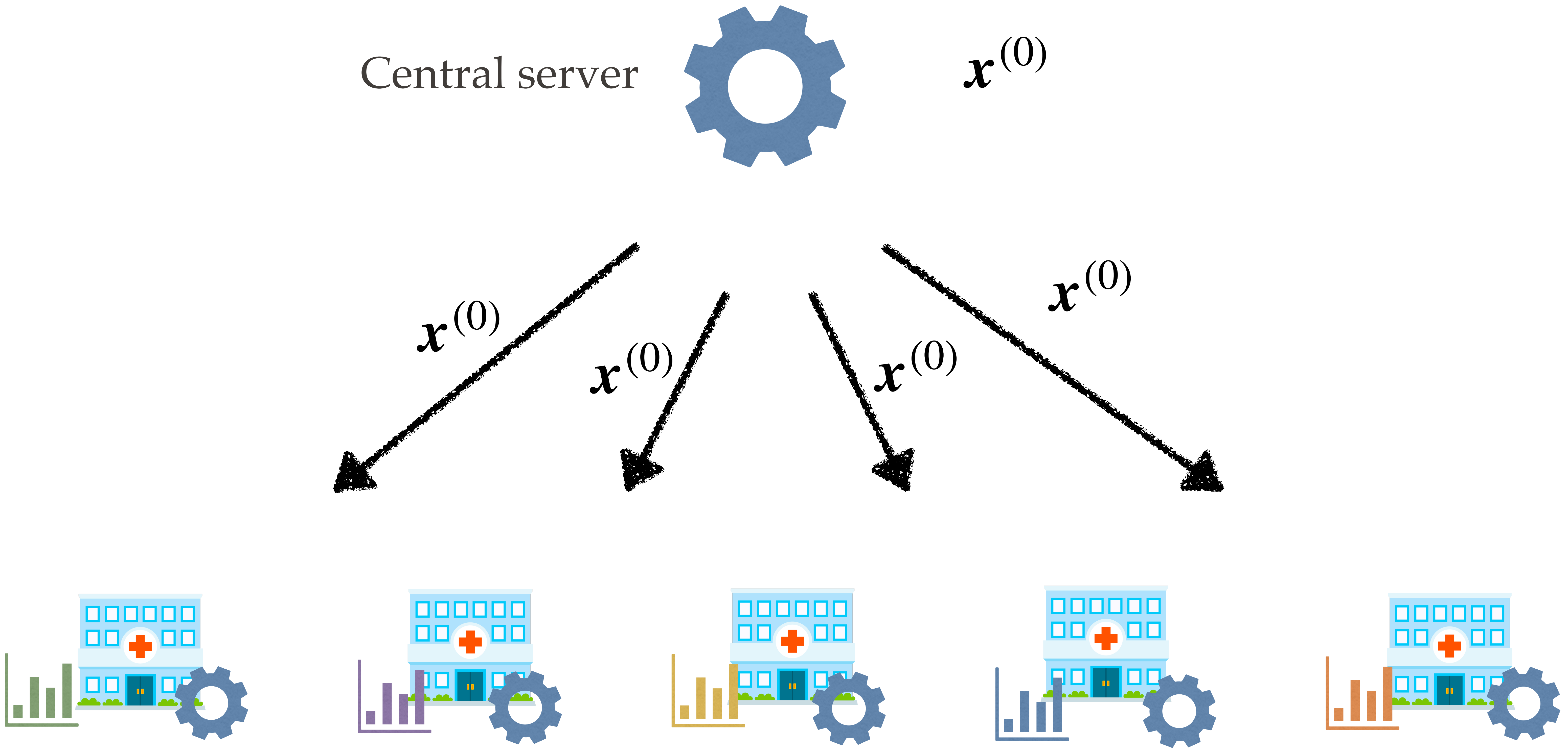
(Abadi et al, 2016)

Central server  $\boldsymbol{x}^{(0)}$

Sever choses the model architecture, and initialises it

FedAvg with Differential Privacy

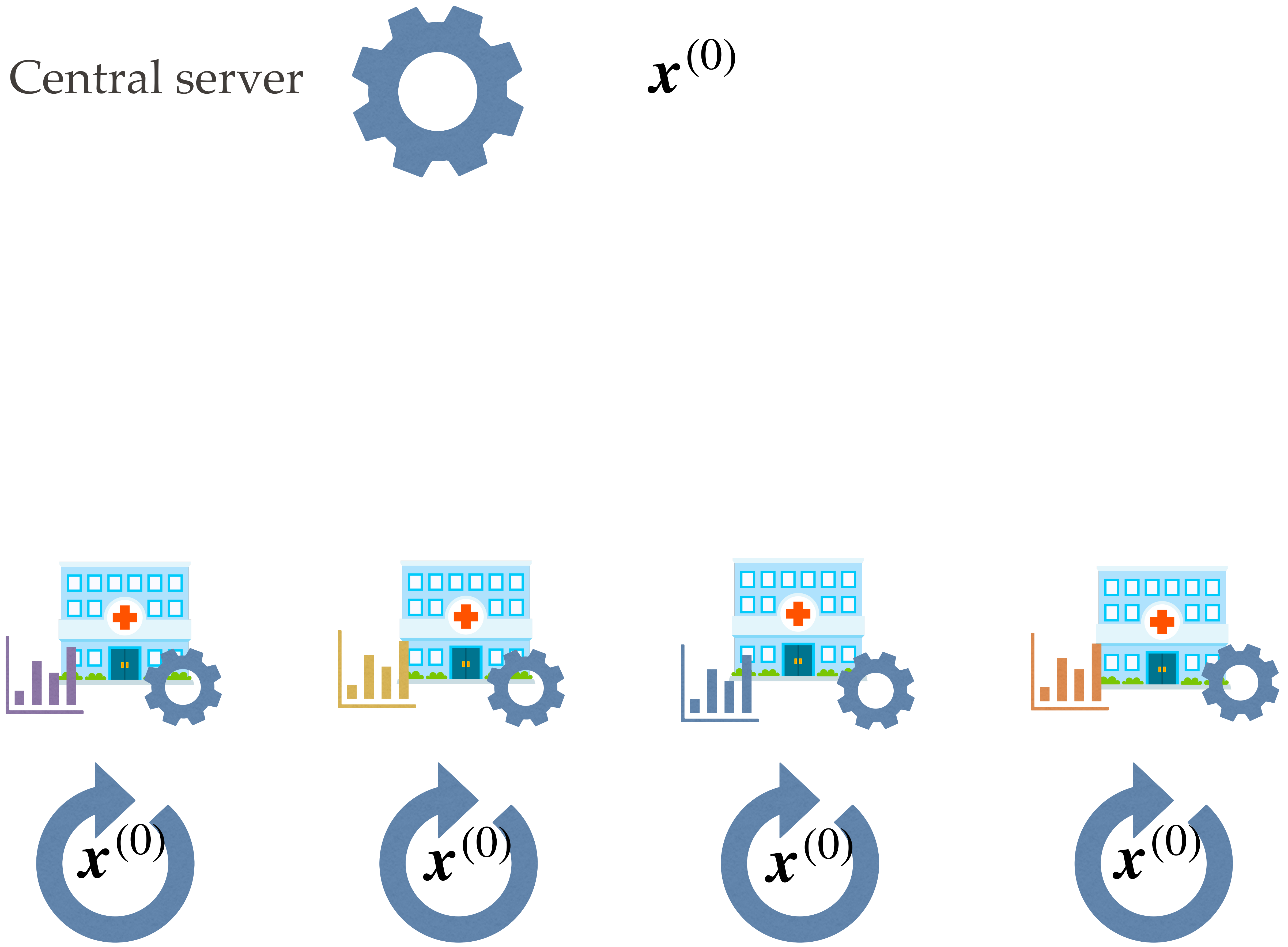
(Abadi et al, 2016)



Sends this model to all the participants

FedAvg with Differential Privacy

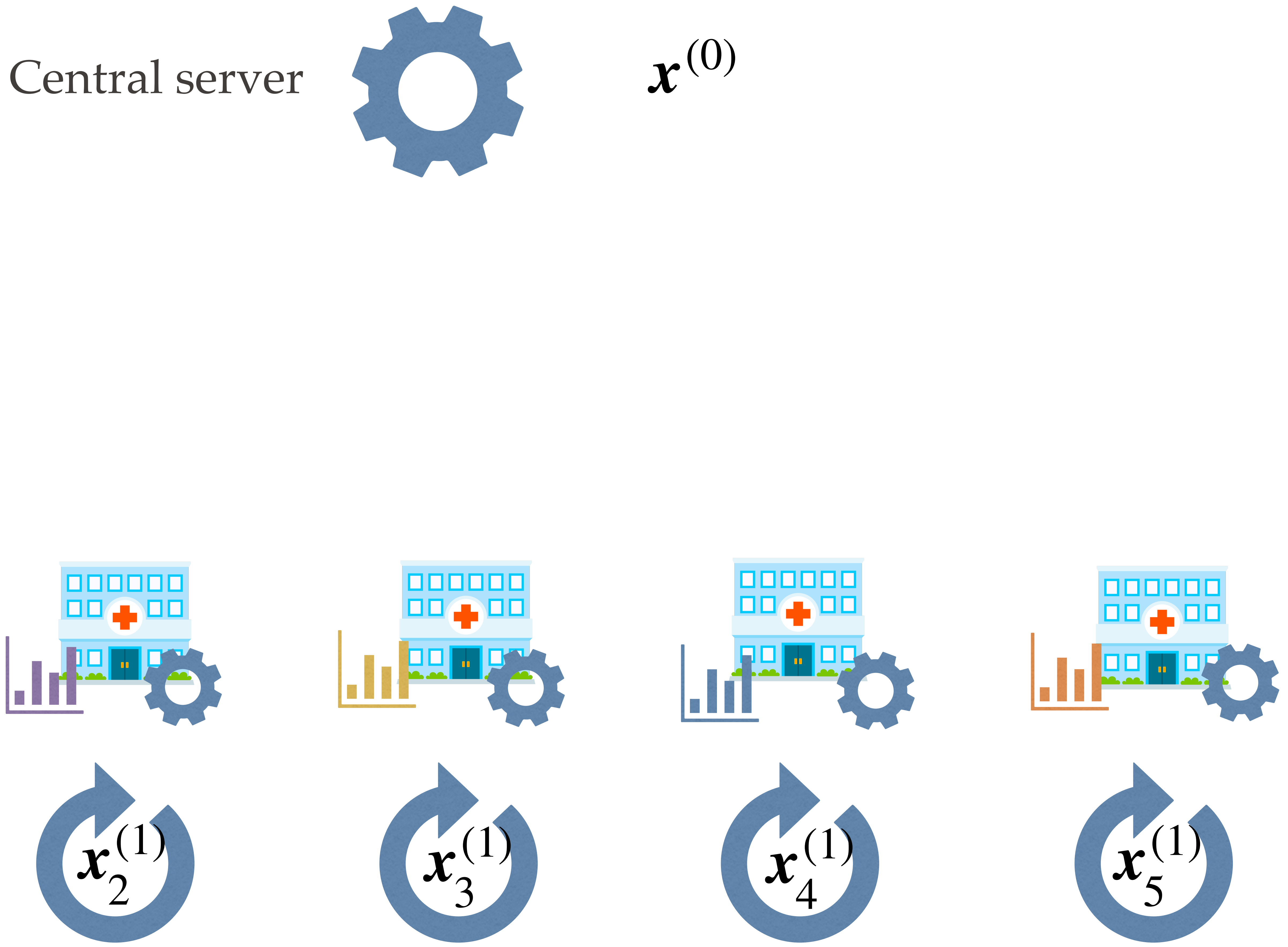
(Abadi et al, 2016)



Clients are performing local update steps based on the local data

FedAvg with Differential Privacy

(Abadi et al, 2016)

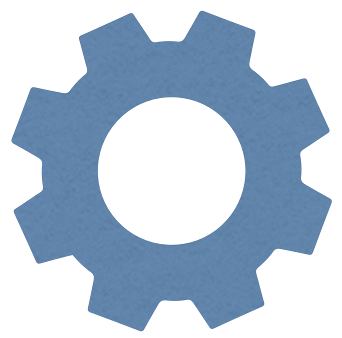


Clients are performing local update steps based on the local data

FedAvg with Differential Privacy

(Abadi et al, 2016)

Central server



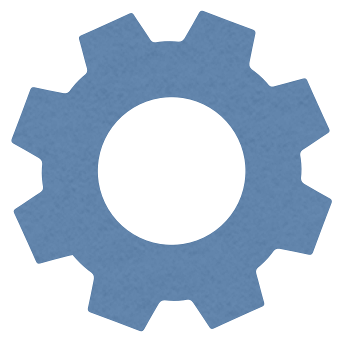
$\boldsymbol{x}^{(0)}$

$clip(\boldsymbol{x}_1^{(1)} - \boldsymbol{x}^{(0)})$ $clip(\boldsymbol{x}_2^{(1)} - \boldsymbol{x}^{(0)})$ $clip(\boldsymbol{x}_3^{(1)} - \boldsymbol{x}^{(0)})$ $clip(\boldsymbol{x}_4^{(1)} - \boldsymbol{x}^{(0)})$ $clip(\boldsymbol{x}_5^{(1)} - \boldsymbol{x}^{(0)})$

Clip the local updates

FedAvg with Differential Privacy

(Abadi et al, 2016)

Central server  $\boldsymbol{x}^{(0)}$

$clip(\boldsymbol{x}_1^{(1)} - \boldsymbol{x}^{(0)})$
$+\mathcal{N}(0,\alpha)$

$clip(\boldsymbol{x}_2^{(1)} - \boldsymbol{x}^{(0)})$
$+\mathcal{N}(0,\alpha)$

$clip(\boldsymbol{x}_3^{(1)} - \boldsymbol{x}^{(0)})$
$+\mathcal{N}(0,\alpha)$

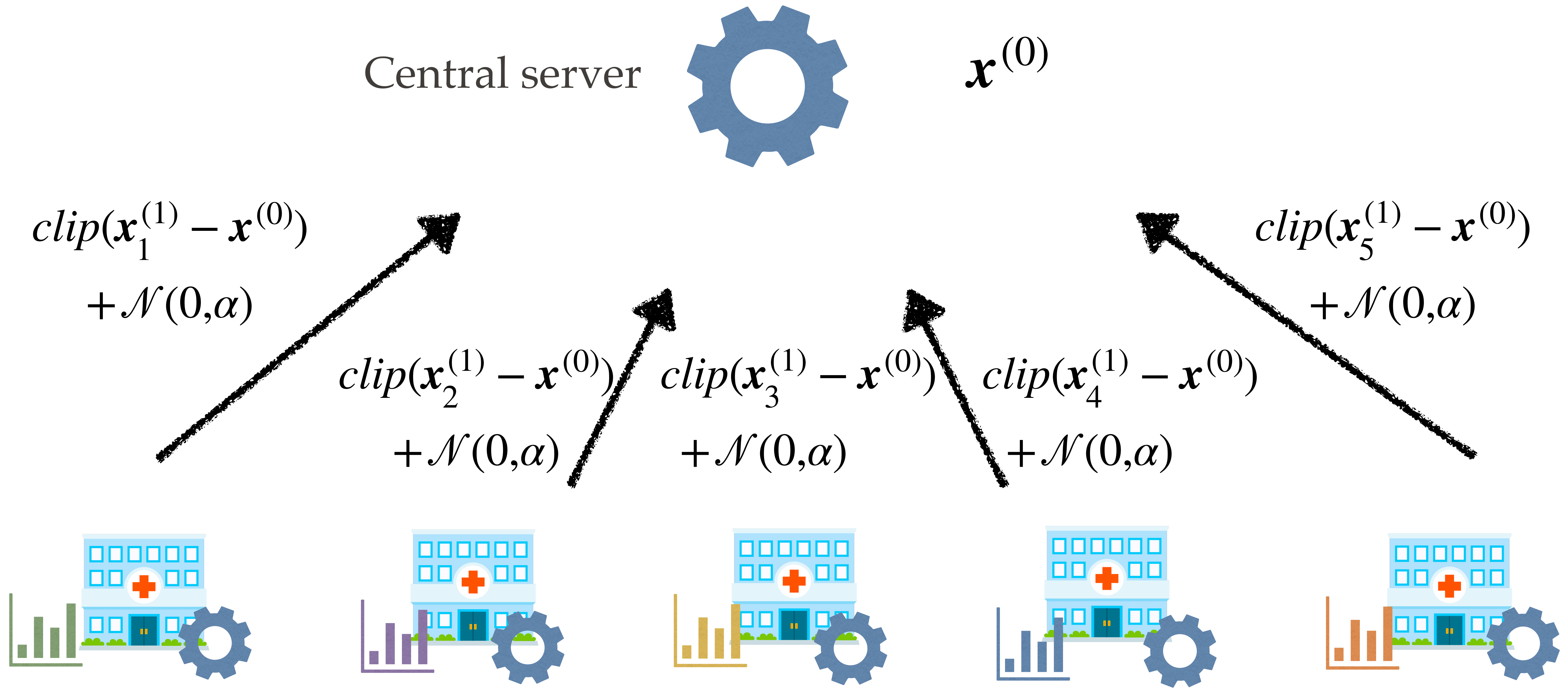
$clip(\boldsymbol{x}_4^{(1)} - \boldsymbol{x}^{(0)})$
$+\mathcal{N}(0,\alpha)$

$clip(\boldsymbol{x}_5^{(1)} - \boldsymbol{x}^{(0)})$
$+\mathcal{N}(0,\alpha)$

Add the privacy noise

FedAvg with Differential Privacy

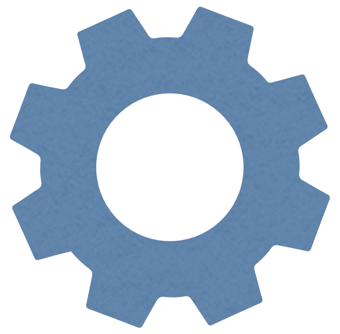
(Abadi et al, 2016)



Send updated models to the server

FedAvg with Differential Privacy

(Abadi et al, 2016)

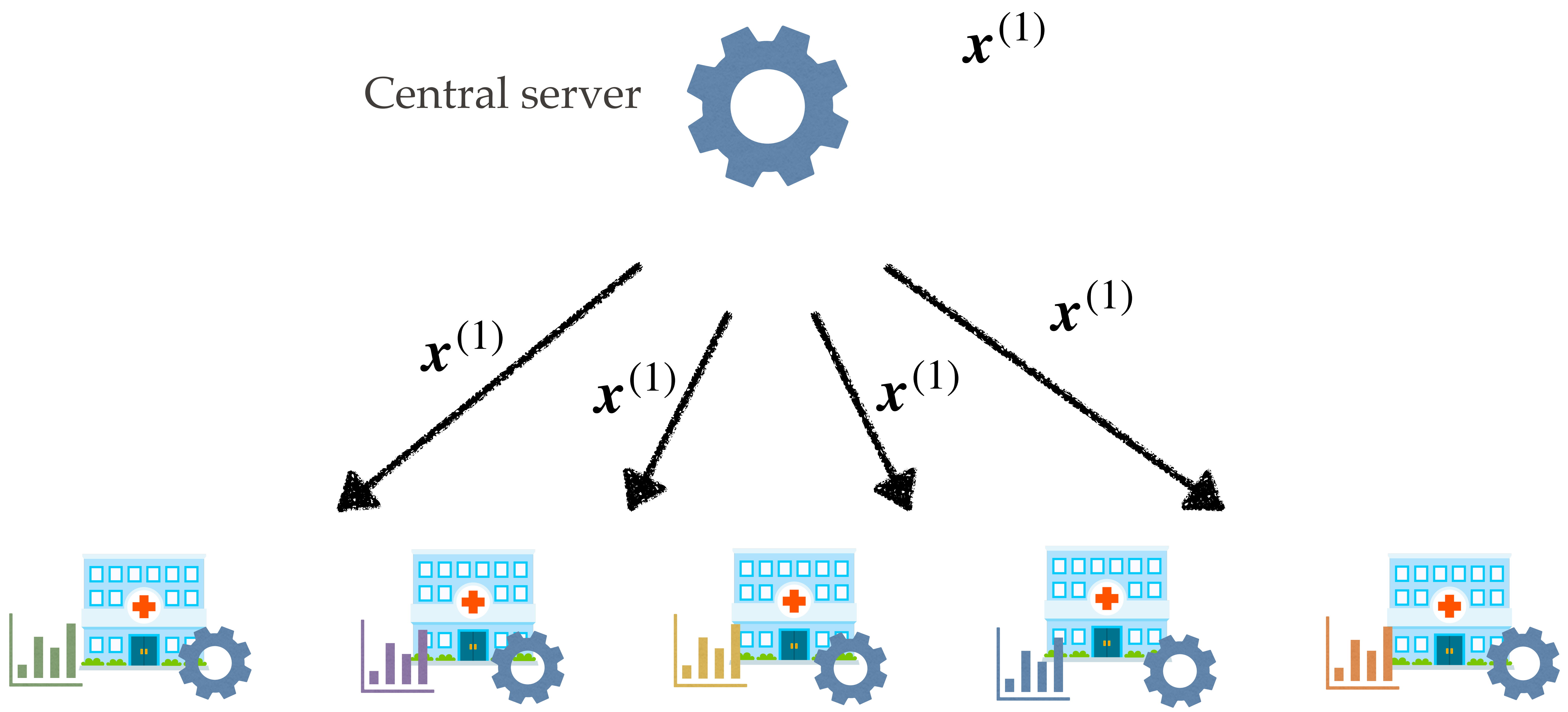
Central server 

$$\boldsymbol{x}^{(1)} = x^{(0)} + \gamma \frac{1}{n} \sum_{i=1}^{n} \left(clip(\boldsymbol{x}_i^{(1)} - \boldsymbol{x}^{(0)}) + \mathcal{N}(0, \alpha) \right)$$

Server averages the updates & updates the global model

FedAvg with Differential Privacy

(Abadi et al, 2016)



Procedure continues for many rounds

Privacy-Utility Tradeoff

The large the noise, the stronger the privacy

The large the noise, the worse the final model performance

Privacy-Utility Tradeoff

The large the noise, the stronger the privacy

The large the noise, the worse the final model performance

Are there the noise distributions that improve privacy
but do not destroy the model performance ?

Challenges in Federated Learning

Data heterogeneity

Local data are different

Communication is slow

Hundreds of MB per model

Privacy

Frequently local data are sensitive & protected by privacy laws

Other Challenges in FL

Malicious or unreliable participants

System heterogeneity

Different participants might have different computing resources

Incentives to participate

References

Konecny, J., McMahan, H. B., Ramage, D., and Richtarik, P. Federated optimization: Distributed machine learning for on-device intelligence. 2016

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. 2017

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, Milan Vojnovic. QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding, 2017

Sebastian U. Stich, Jean-Baptiste Cordonnier, Martin Jaggi. Sparsified SGD with Memory, 2018

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of FedAvg on non-iid data. 2018

S.P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, A. T. Suresh. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning, 2019

T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, V. Smith. Federated Optimization in Heterogeneous Networks, 2019

References

Alireza Fallah, Aryan Mokhtari, Asuman Ozdaglar. Personalized Federated Learning: A Meta-Learning Approach, 2020

F. Chen, M. Luo, Z. Dong, Z. Li, and X. He, “Federated meta-learning with fast convergence and efficient communication, 2018

C. Dwork, F. McSherry, K. Nissim & A. Smith, Calibrating Noise to Sensitivity in Private Data Analysis, 2006

Martín Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, Li Zhang, Deep Learning with Differential Privacy, 2016

Peter Kairouz, H. Brendan McMahan, et al. Advances and Open Problems in Federated Learning, 2019