DMQA Open Seminar

Federated Learning

2025. 03. 07

Korea University

Data **M**ining & **Q**uality **A**nalytics Lab.

최지형

Data Mining
Quality Analytics

발표자 소개

❖ **최지형 (Jihyung Choi)**

- 고려대학교 산업경영공학과 대학원 재학
- Data Mining & Quality Analytics Lab. (김성범 교수님)
- M.S Student (2024.09 ~ Present)

❖ **Research Interest**

- Federated Learning
- Fine-tuning Foundation Models
- Agent AI

❖ **Contact**

- jibro@korea.ac.kr

Introduction

What is Federated Learning?

❖ Federated Learning (FL)

Q. 데이터가 부족한가?

"우리 집 고양이 츄르를 좋아해."

Introduction

What is Federated Learning?

❖ Federated Learning (FL)

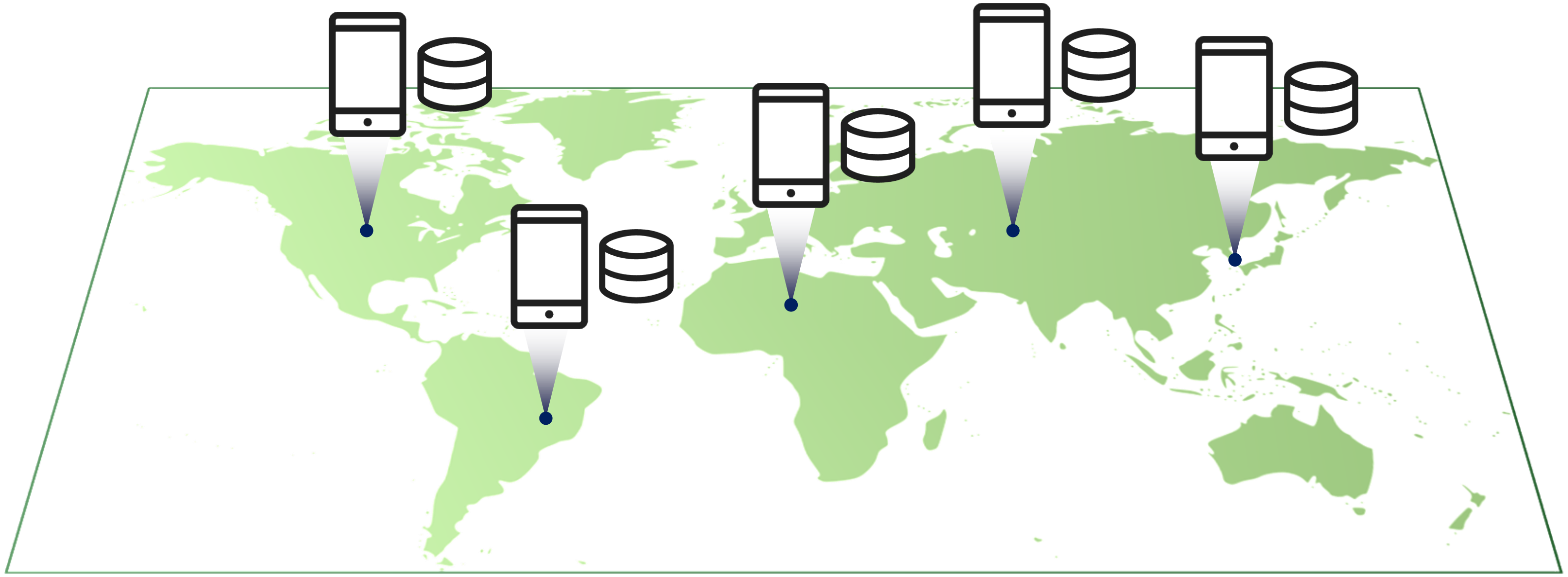
Q. 데이터가 부족한가?

Introduction

What is Federated Learning?

❖ Federated Learning (FL)

Q. 데이터가 부족한가?

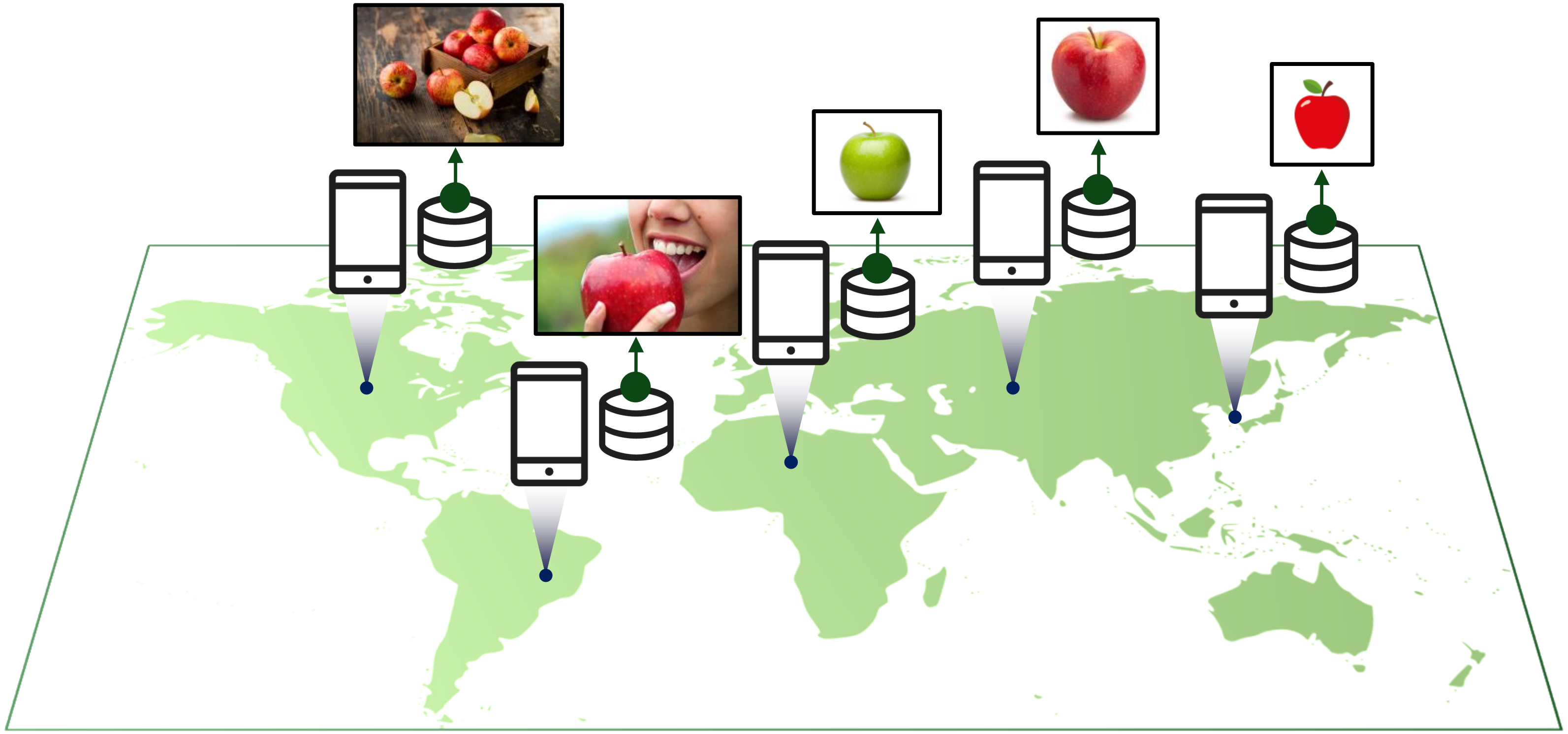


Introduction

What is Federated Learning?

❖ Federated Learning (FL)

Q. 데이터가 부족한가?



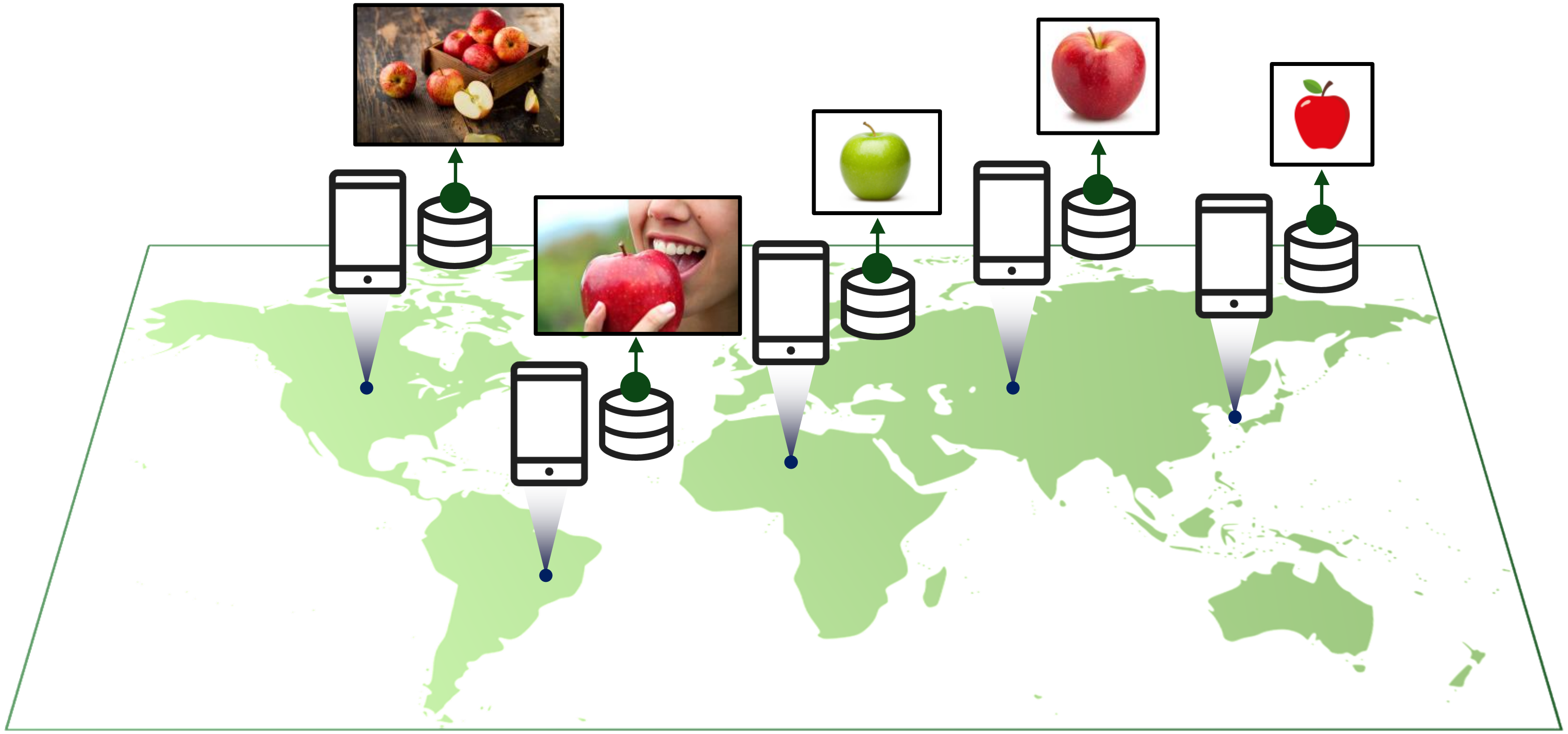
Introduction

What is Federated Learning?

❖ Federated Learning (FL)

Q. 데이터가 부족한가?

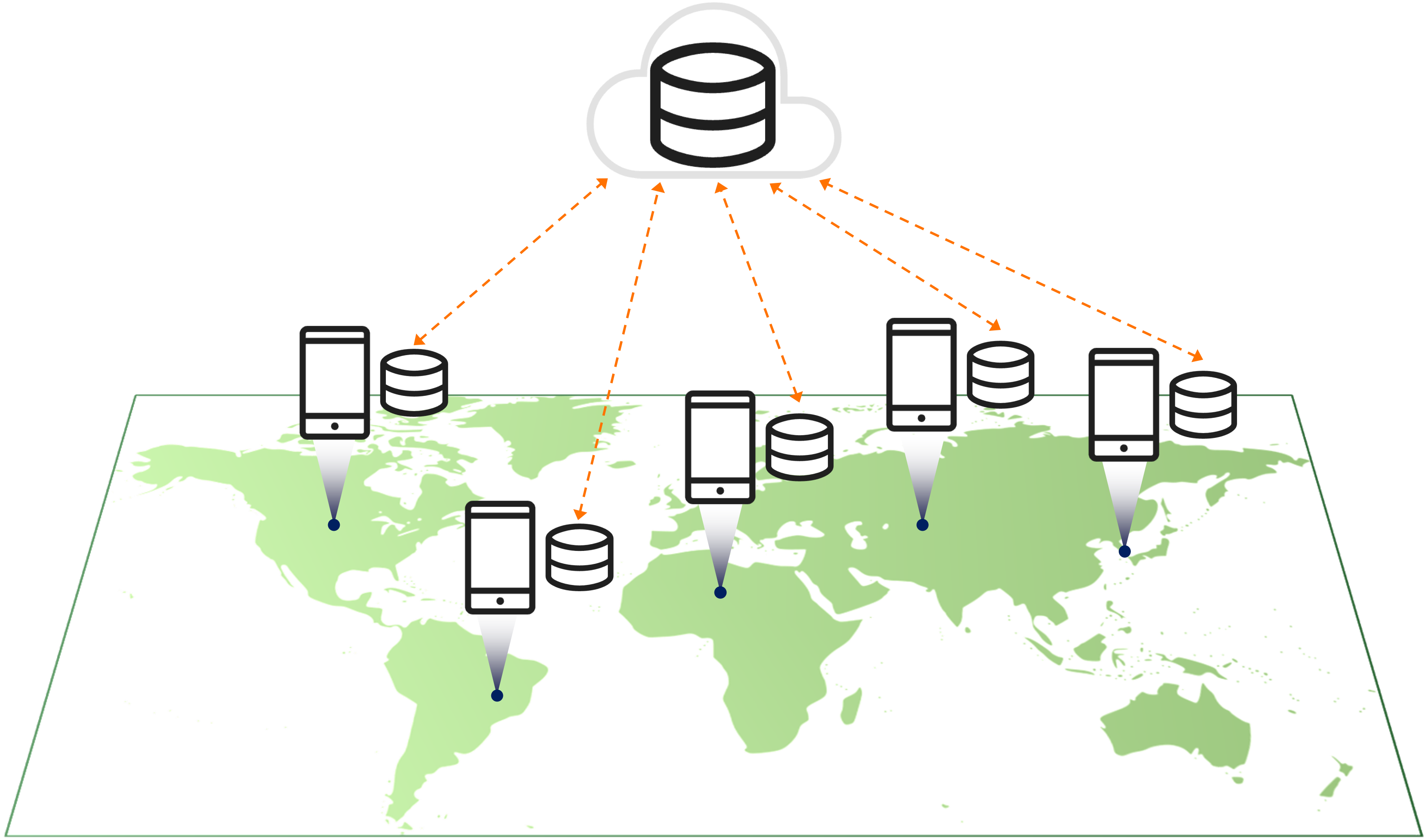
A. 데이터 활용 방법이 부족하다!



Introduction

What is Federated Learning?

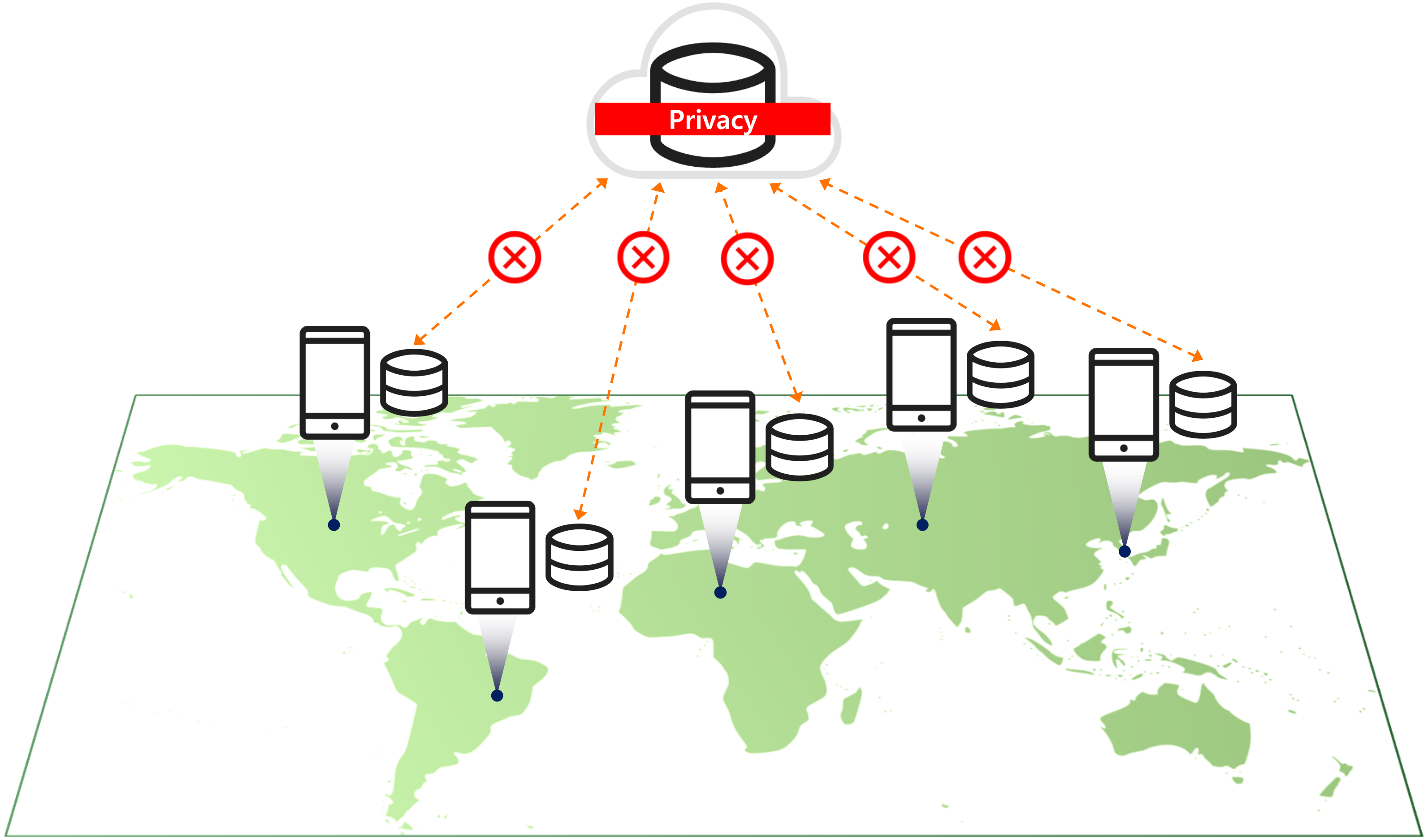
❖ **Federated Learning (FL)**



Introduction

What is Federated Learning?

❖ Federated Learning (FL)

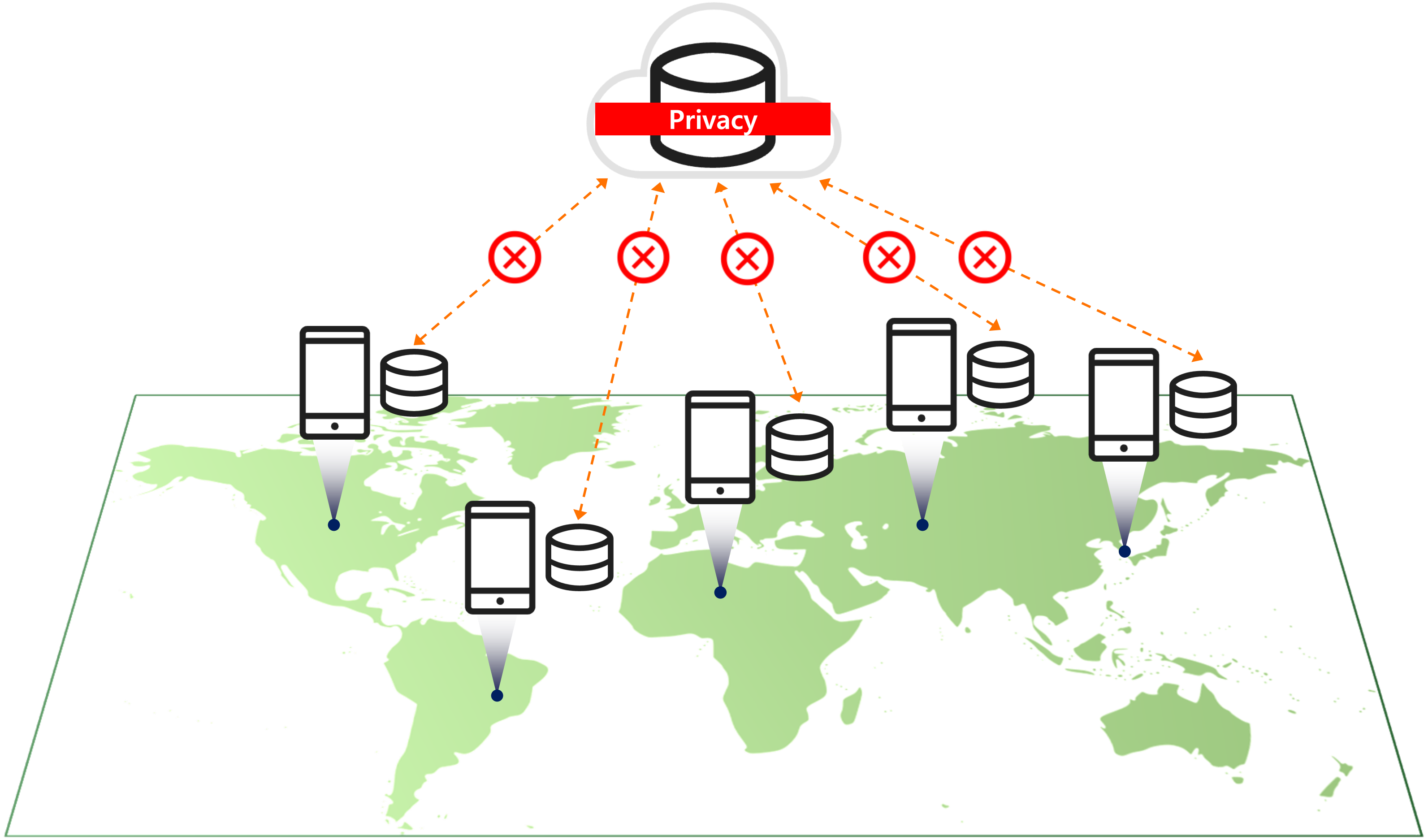


Introduction

What is Federated Learning?

❖ **Federated Learning (FL)**

➢ **데이터가 분산된 환경에, 데이터 공유 없이 모델 학습!!**



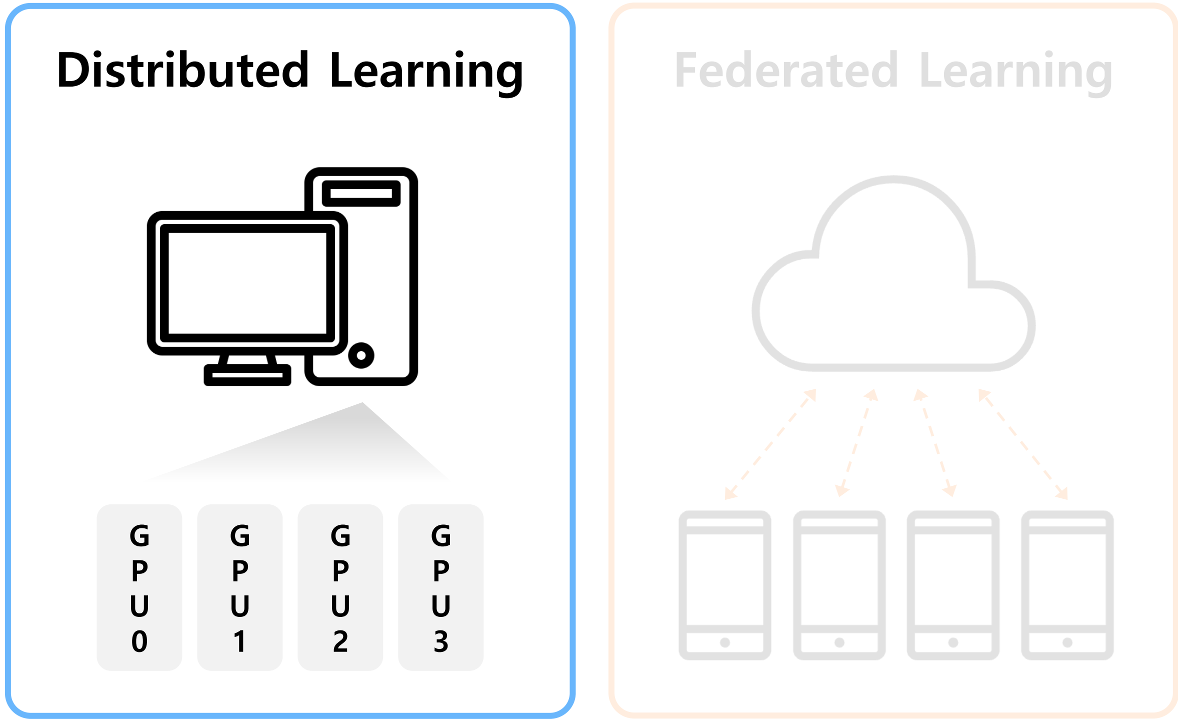
Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O



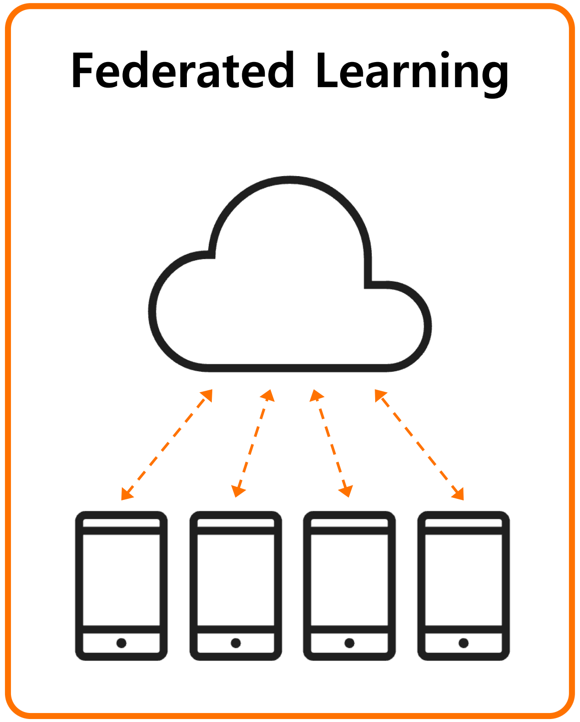
Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ 데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O



Introduction

Federated Learning vs. Distributed Learning

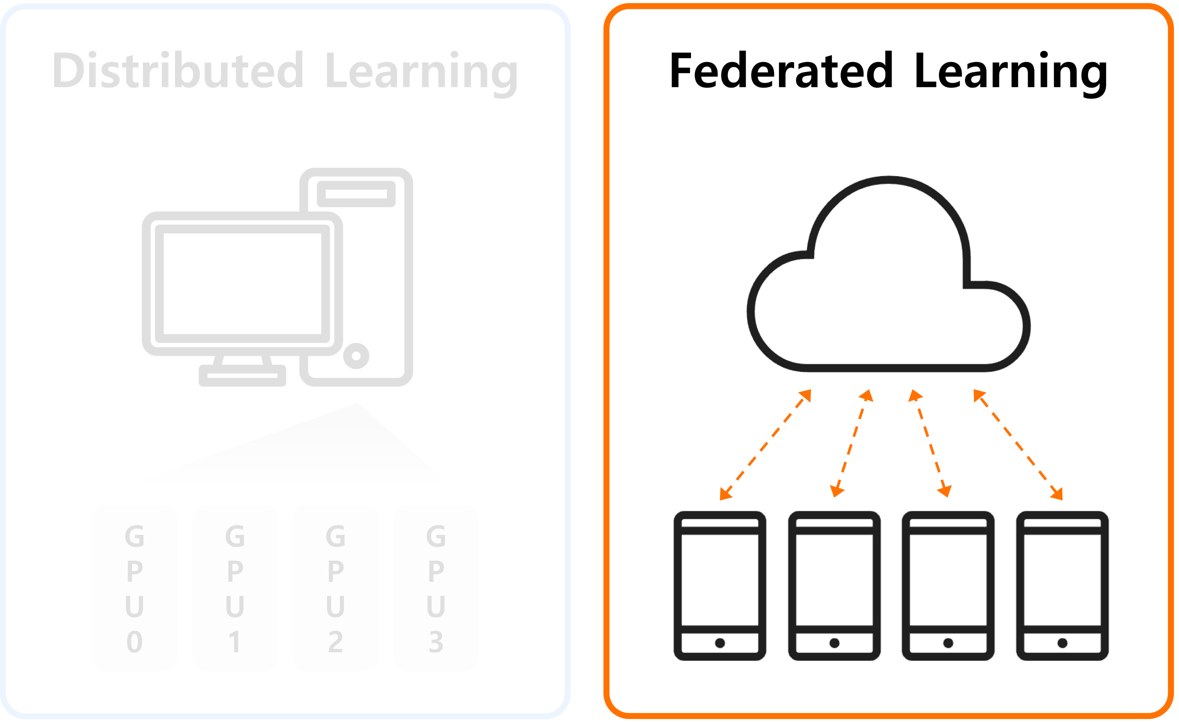
❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O

communication cost

partial participation

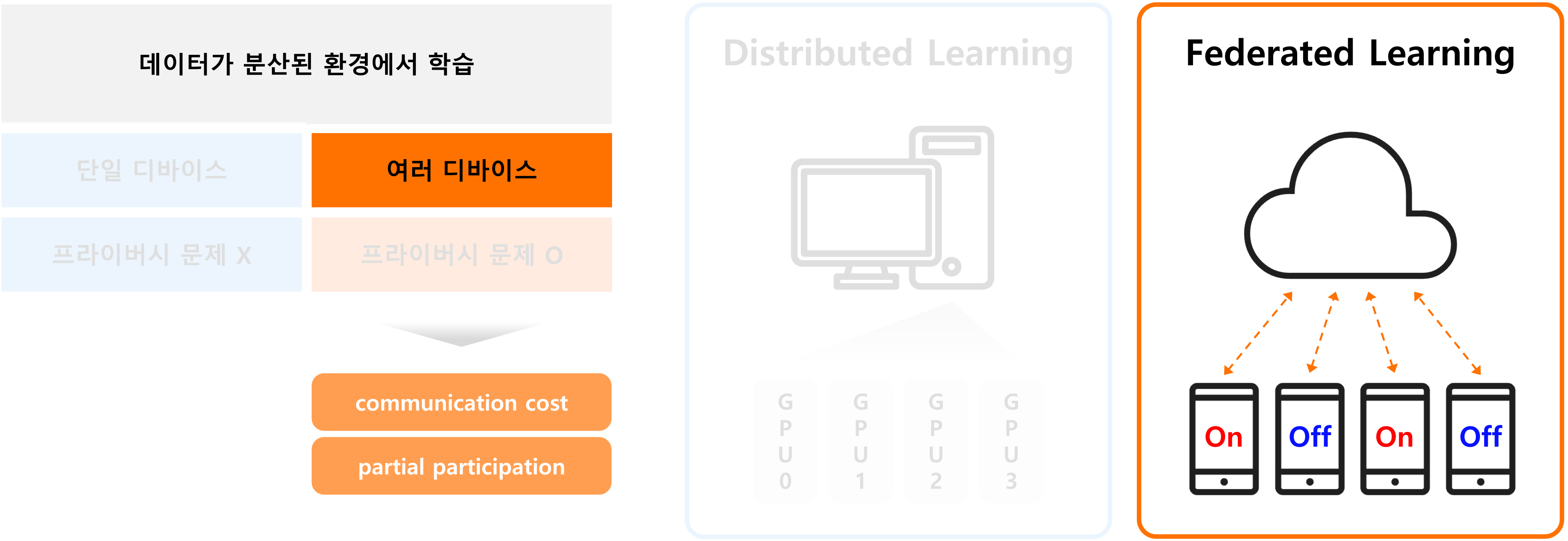


Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**



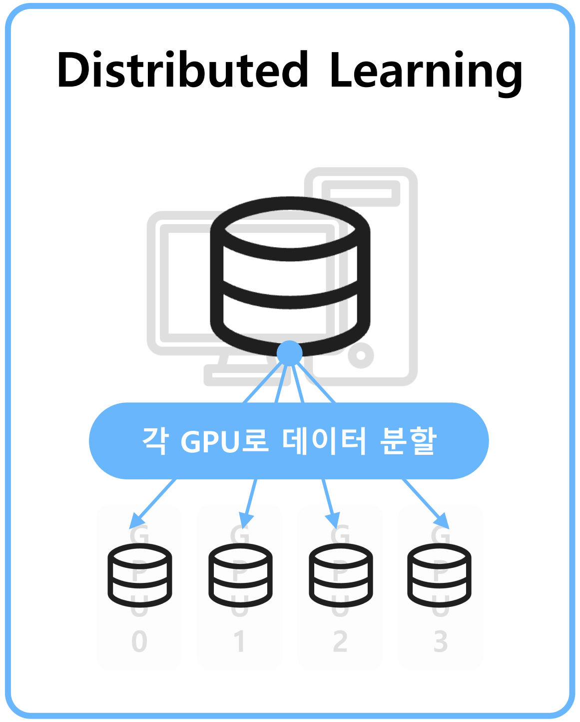
Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O



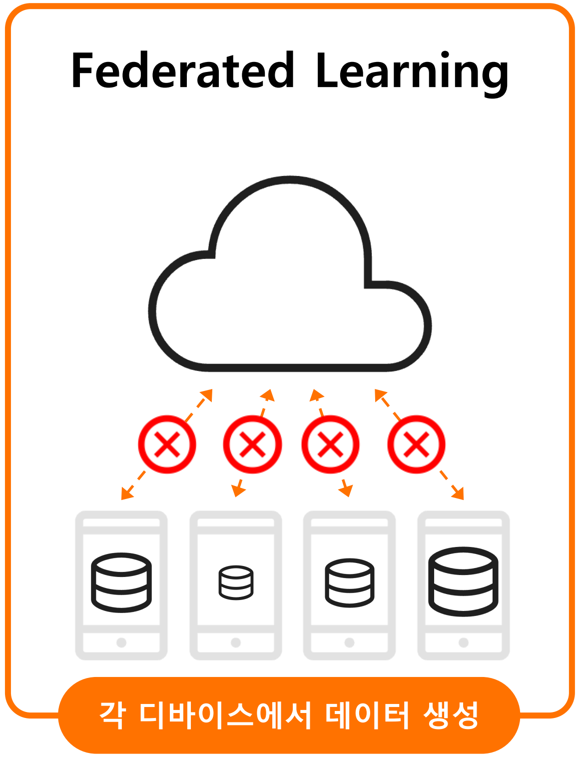
Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O
	communication cost
	partial participation



Introduction

Federated Learning vs. Distributed Learning

❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습

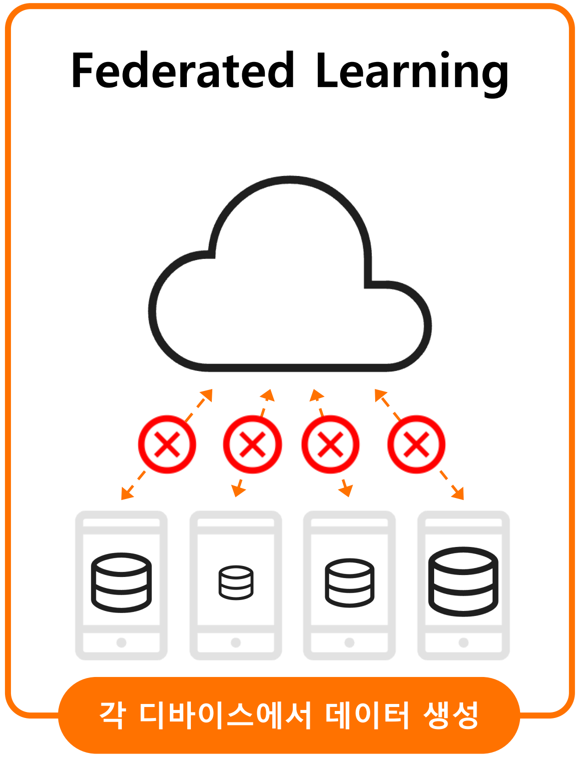
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O

communication cost

partial participation

Non-I.I.D

데이터 불균형



Introduction

Federated Learning vs. Distributed Learning

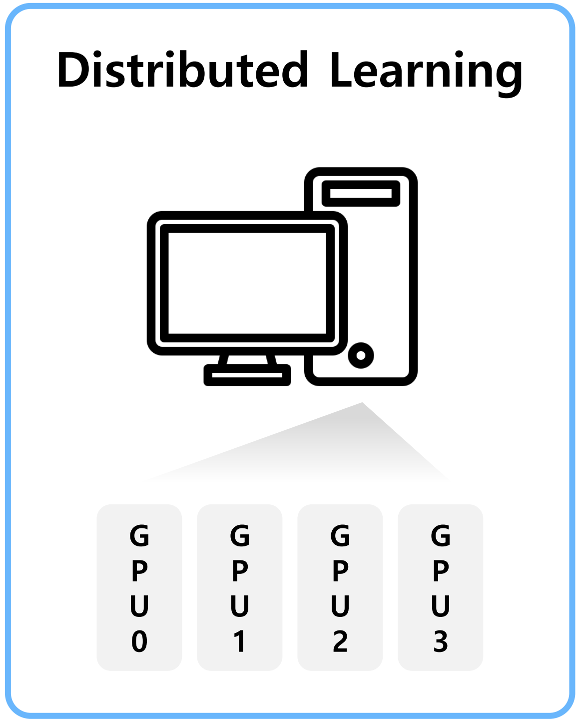
❖ **Distributed Learning:** 데이터를 여러 연산 장치로 나누어 학습

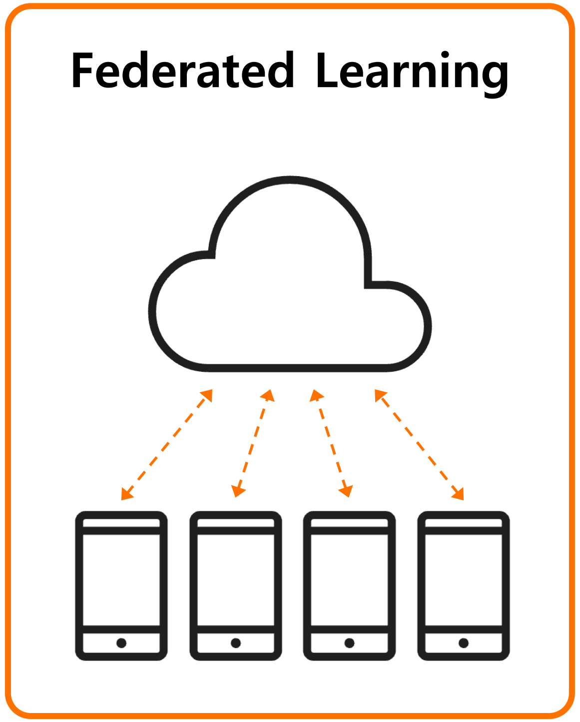
➢ **데이터가 분산된 환경에서 학습한다는 점에서, FL과 동일**

데이터가 분산된 환경에서 학습	
단일 디바이스	여러 디바이스
프라이버시 문제 X	프라이버시 문제 O

FL이 해결해야 하는 4가지 문제

- communication cost
- partial participation
- Non-I.I.D
- 데이터 불균형



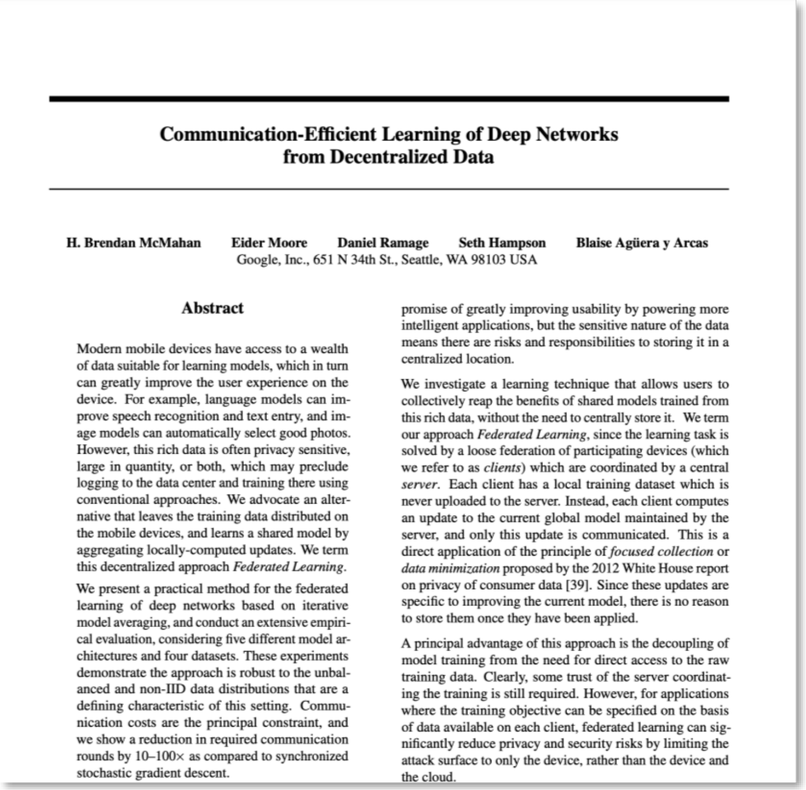


Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ **Federated learning 개념과 federated averaging 알고리즘 제안**

- AISTATS'17
- 피인용 21686회 (2025년 3월 기준)



Communication-Efficient Learning of Deep Networks from Decentralized Data

H. Brendan McMahan Eider Moore Daniel Ramage Seth Hampson Blaise Agüera y Arcas
Google, Inc., 651 N 34th St., Seattle, WA 98103 USA

Abstract

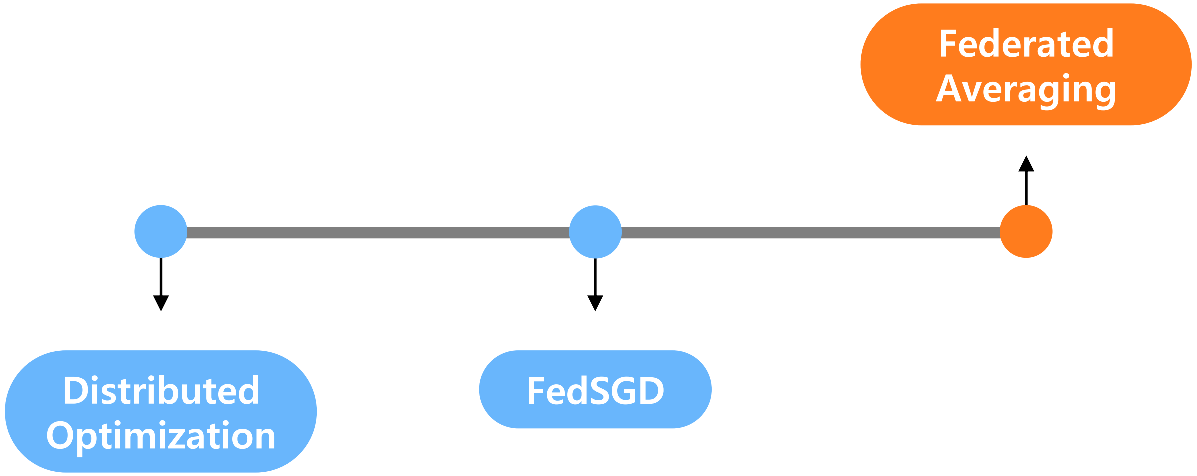
Modern mobile devices have access to a wealth of data suitable for learning models, which in turn can greatly improve the user experience on the device. For example, language models can improve speech recognition and text entry, and image models can automatically select good photos. However, this rich data is often privacy sensitive, large in quantity, or both, which may preclude logging to the data center and training there using conventional approaches. We advocate an alternative that leaves the training data distributed on the mobile devices, and learns a shared model by aggregating locally-computed updates. We term this decentralized approach *Federated Learning*.

We present a practical method for the federated learning of deep networks based on iterative model averaging, and conduct an extensive empirical evaluation, considering five different model architectures and four datasets. These experiments demonstrate the approach is robust to the unbalanced and non-IID data distributions that are a defining characteristic of this setting. Communication costs are the principal constraint, and we show a reduction in required communication rounds by 10–100× as compared to synchronized stochastic gradient descent.

promise of greatly improving usability by powering more intelligent applications, but the sensitive nature of the data means there are risks and responsibilities to storing it in a centralized location.

We investigate a learning technique that allows users to collectively reap the benefits of shared models trained from this rich data, without the need to centrally store it. We term our approach *Federated Learning*, since the learning task is solved by a loose federation of participating devices (which we refer to as *clients*) which are coordinated by a central *server*. Each client has a local training dataset which is never uploaded to the server. Instead, each client computes an update to the current global model maintained by the server, and only this update is communicated. This is a direct application of the principle of *focused collection* or *data minimization* proposed by the 2012 White House report on privacy of consumer data [39]. Since these updates are specific to improving the current model, there is no reason to store them once they have been applied.

A principal advantage of this approach is the decoupling of model training from the need for direct access to the raw training data. Clearly, some trust of the server coordinating the training is still required. However, for applications where the training objective can be specified on the basis of data available on each client, federated learning can significantly reduce privacy and security risks by limiting the attack surface to only the device, rather than the device and the cloud.



McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Agüera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, JMLR: W&CP, 54.

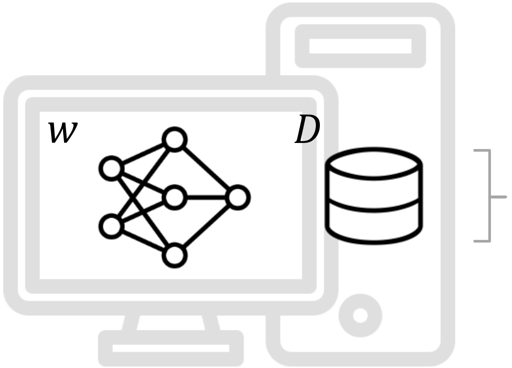
Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$



Data size: n

Total Loss: $f(w) := \frac{1}{n}\sum f_i(w)$

GPU: 1 GPU: 2 GPU: 3 … GPU: K

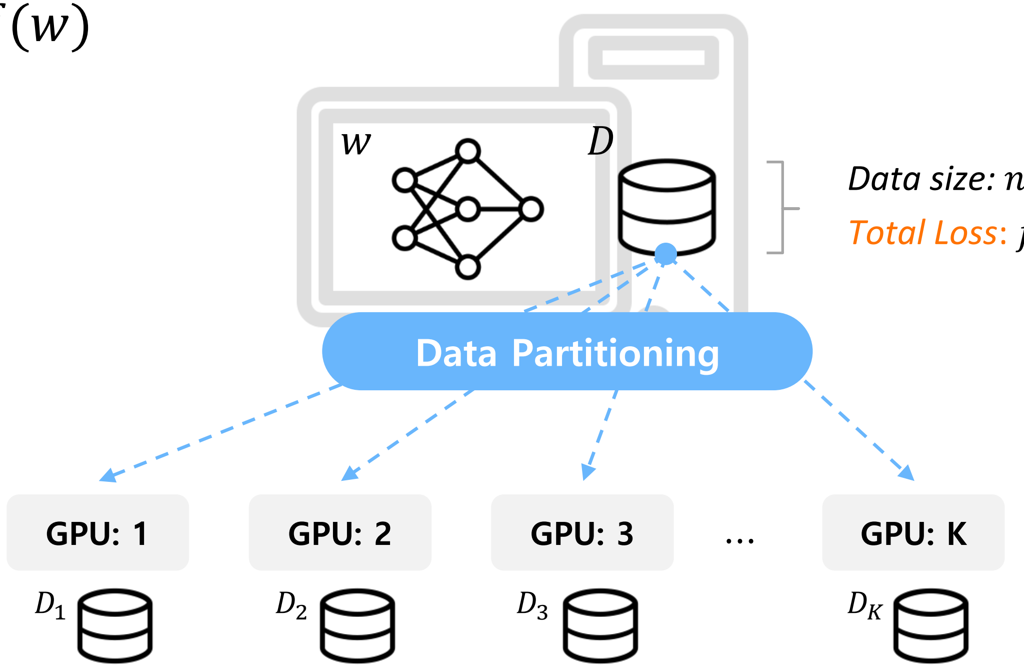
Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$



Data size: n

Total Loss: $f(w) \coloneqq \frac{1}{n}\sum f_i(w)$

Data size: $\frac{n}{K}$

Partial Loss: $F_k(w) = \frac{K}{n}\sum_{i \in D_k} f_i(w)$

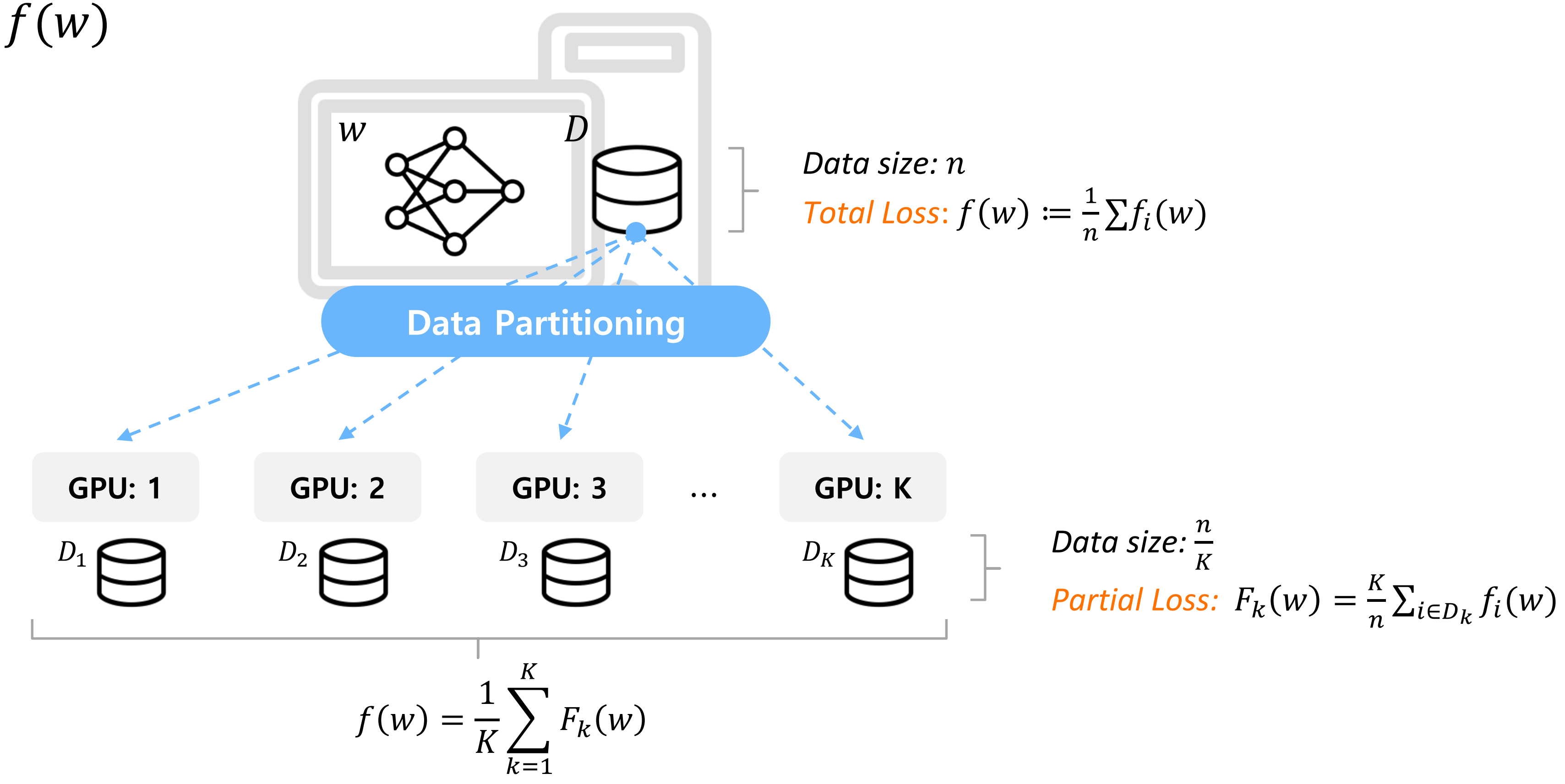
Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w_t)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w_t)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^{k}$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

$$= w_t - \eta \nabla \{ \frac{1}{K} \sum_{k=1}^{K} F_k(w_t) \}$$

$$f(w) = \frac{1}{K} \sum_{k=1}^{K} F_k(w)$$

$$= w_t - \frac{1}{K} \sum_{k=1}^{K} \eta \nabla F_k(w_t)$$

$$= \frac{1}{K} \sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

$$= \frac{1}{K} \sum_{k=1}^{K} w_{t+1}^{k}$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$

Total Loss Gradient

$$f(w) = \frac{1}{K}\sum_{k=1}^{K} F_k(w)$$

$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w_t)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w_t)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^{k}$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

$$\begin{aligned} w_{t+1} &= w_t - \eta \nabla f(w_t) \quad \text{Total Loss Gradient} \\ &= w_t - \eta \nabla \{ \frac{1}{K} \sum_{k=1}^{K} F_k(w_t) \} \\ &= w_t - \frac{1}{K} \sum_{k=1}^{K} \eta \nabla F_k(w_t) \\ &= \frac{1}{K} \sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t)) \\ &= \frac{1}{K} \sum_{k=1}^{K} w_{t+1}^{k} \end{aligned}$$

$$f(w) = \frac{1}{K} \sum_{k=1}^{K} F_k(w)$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$

Total Loss Gradient

$$f(w) = \frac{1}{K}\sum_{k=1}^{K} F_k(w)$$

$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

New Gradient Descent???

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^k$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

$$f(w) = \frac{1}{K}\sum_{k=1}^{K} F_k(w)$$

$$= w_t - \eta \nabla\{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$

Partial Loss Gradient

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^{k}$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$

Total Loss Gradient ⇨ D : $w_t \rightarrow w_{t+1}$

$$= w_t - \eta \nabla \{ \frac{1}{K} \sum_{k=1}^{K} F_k(w) \}$$

$$= w_t - \frac{1}{K} \sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K} \sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

Partial Loss Gradient ⇨ D_k : $w_t \rightarrow w_{t+1}^{k}$ *Local Update*

$$= \frac{1}{K} \sum_{k=1}^{K} w_{t+1}^{k}$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_{w} f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$

Total Loss Gradient

D : $w_t \rightarrow w_{t+1}$

$$= w_t - \eta \nabla \{\frac{1}{K} \sum_{k=1}^{K} F_k(w)\}$$

$$= w_t - \frac{1}{K} \sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K} \sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

Partial Loss Gradient

Local Update

D_k : $w_t \rightarrow w_{t+1}^{k}$

$$= \frac{1}{K} \sum_{k=1}^{K} w_{t+1}^{k}$$

글로벌 업데이트 = 로컬 업데이트 수행 후 평균 !!

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

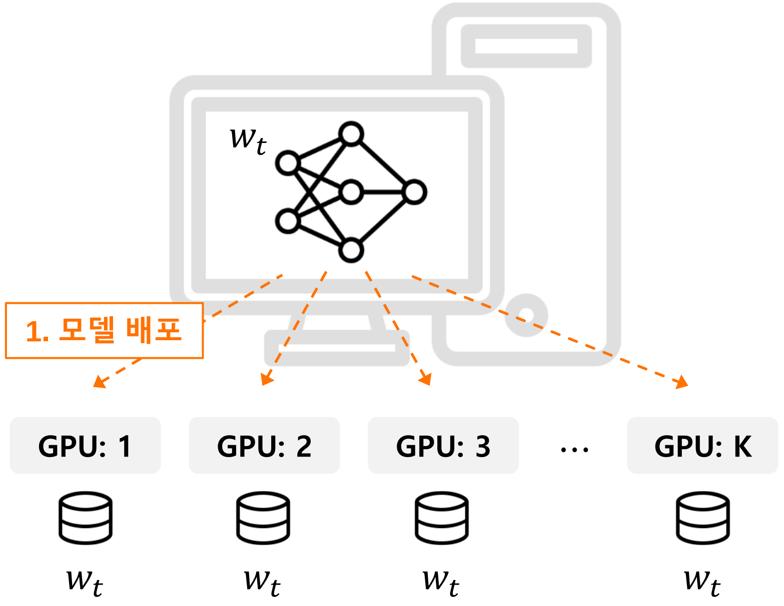
$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

Partial Loss Gradient / *Local Update*

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^k$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal $w^* \triangleq \min_{w} f(w)$

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

$$= w_t - \eta \nabla\{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$

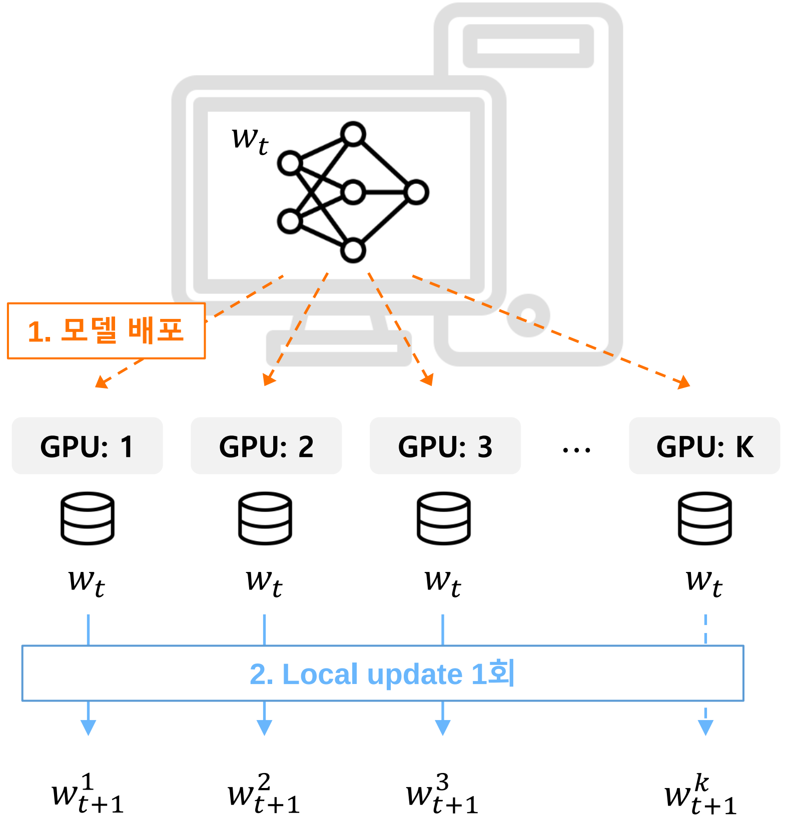
$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$

Partial Loss Gradient

Local Update

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^{k}$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Distributed Optimization

➢ **전체 데이터에 대한 손실함수 값, $f(w)$ 최소화**

Goal | $w^* \triangleq \min_w f(w)$

Gradient Descent

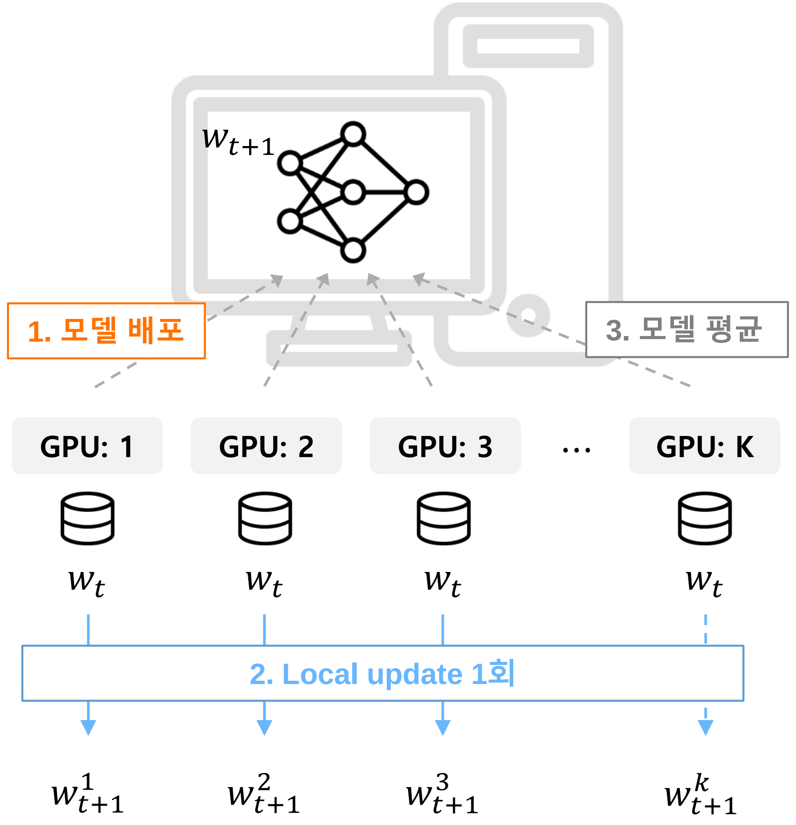
$$w_{t+1} = w_t - \eta \nabla f(w_t)$$ *Total Loss Gradient*

$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$

$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$

$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w_t))$$ *Partial Loss Gradient* / *Local Update*

$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^k$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ **FedSGD**

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

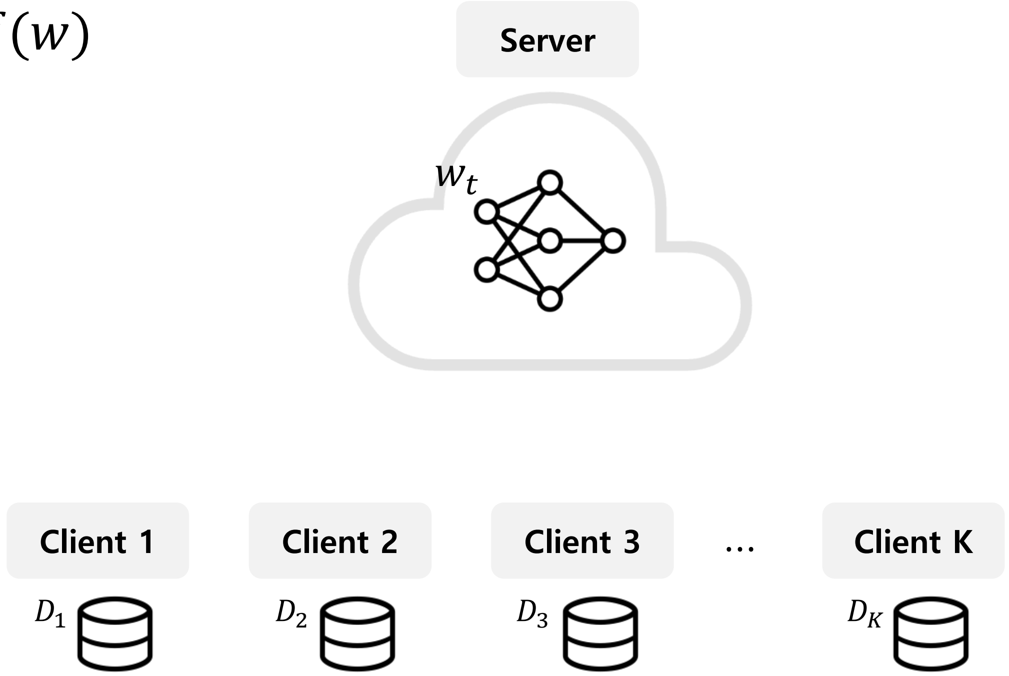
communication cost

partial participation

Non-I.I.D

데이터 불균형

Goal | $w^* \triangleq \min_{w} f(w)$



각 디바이스에서 데이터 생성

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

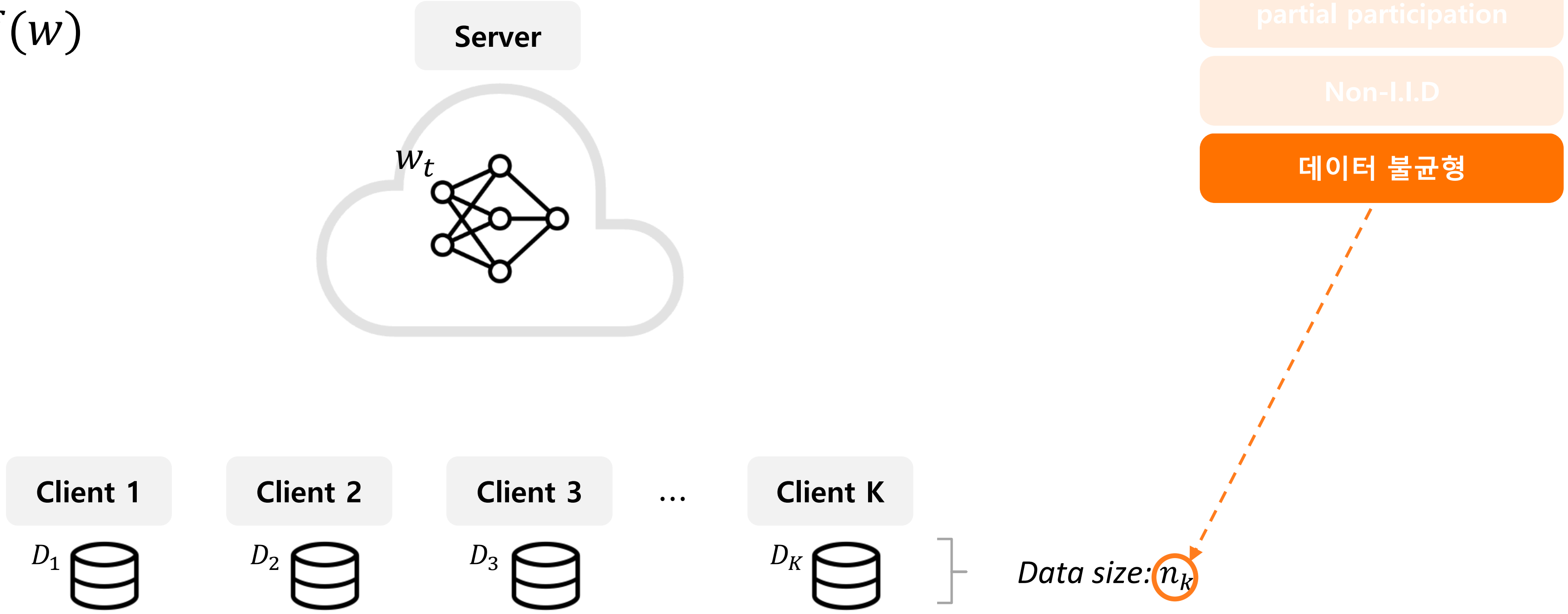
Goal $w^* \triangleq \min_{w} f(w)$

communication cost

partial participation

Non-I.I.D

데이터 불균형



각 디바이스에서 데이터 생성

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

- ➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

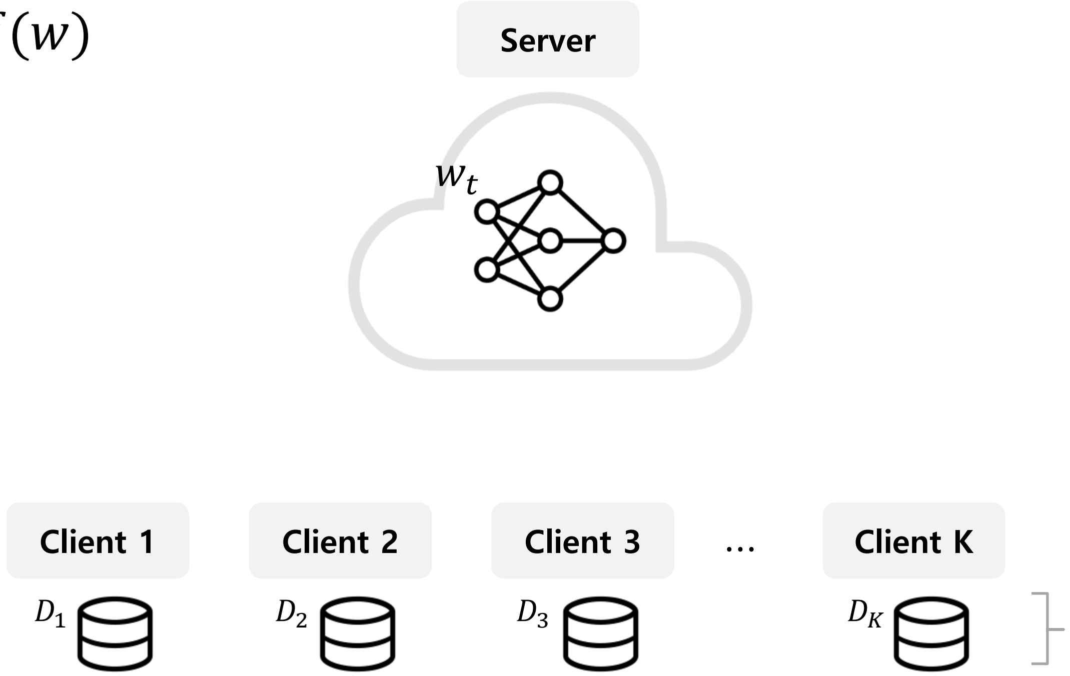
communication cost

partial participation

Non-I.I.D

데이터 불균형

Goal | $w^* \triangleq \min_{w} f(w)$



Data size: n_k

Partial Loss: $F_k(w) = \frac{1}{n_k} \sum_{i \in D_k} f_i(w)$

각 디바이스에서 데이터 생성

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

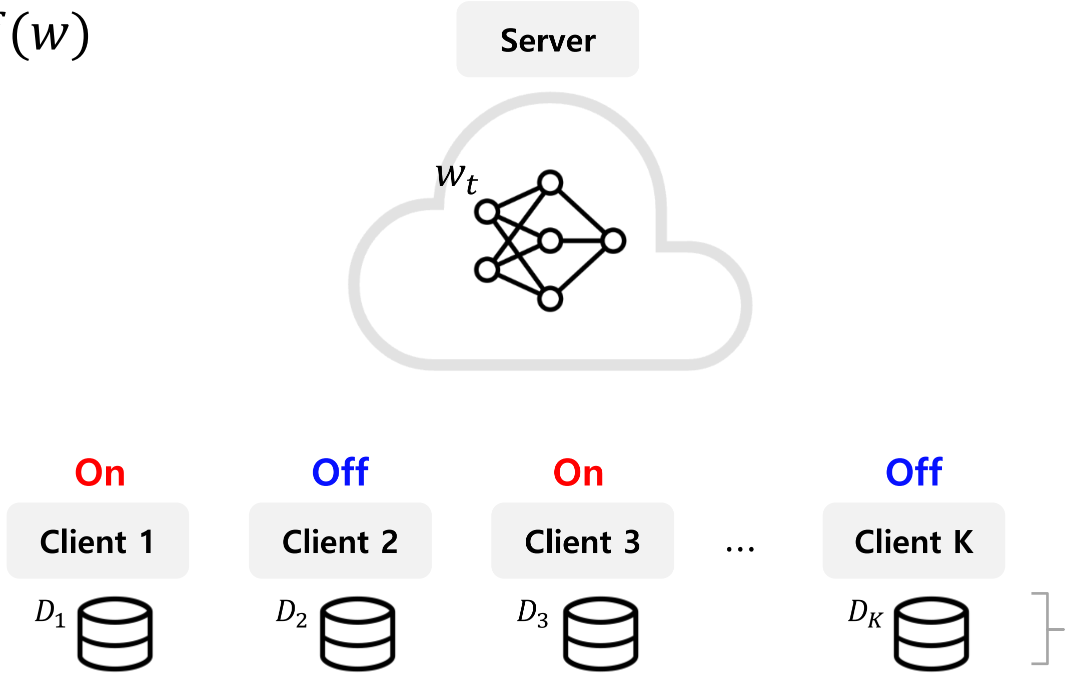
communication cost

partial participation

Non-I.I.D

데이터 불균형

Goal | $w^* \triangleq \min_{w} f(w)$



Data size: n_k

Partial Loss: $F_k(w) = \frac{1}{n_k}\sum_{i \in D_k} f_i(w)$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

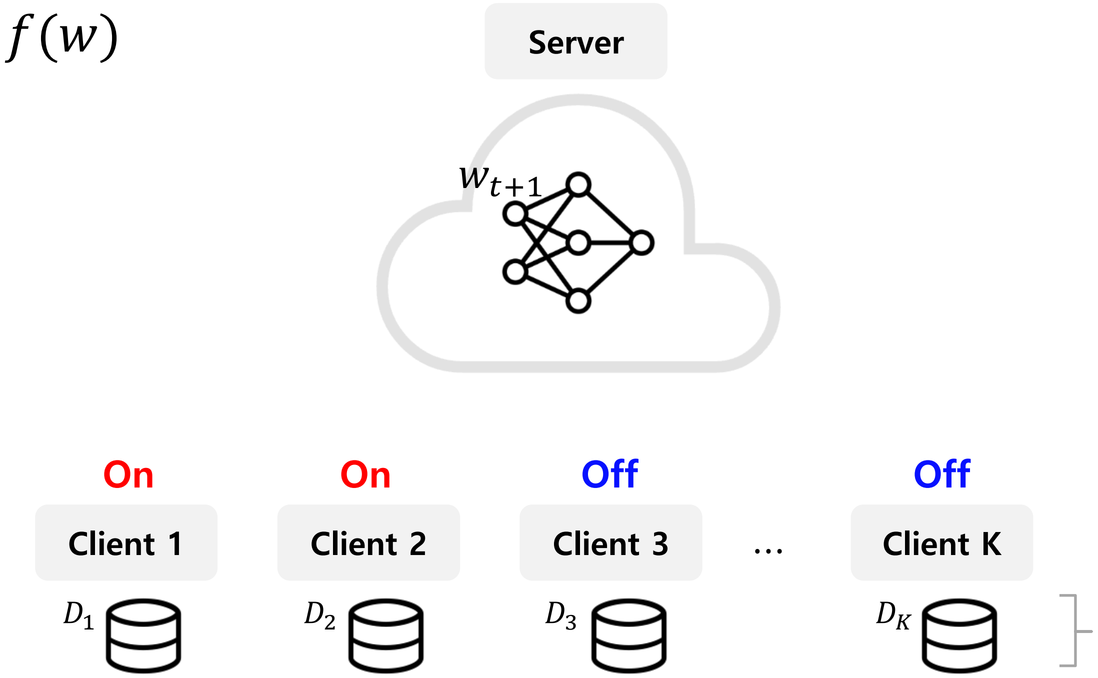
Goal | $w^* \triangleq \min_{w} f(w)$

communication cost

partial participation

Non-I.I.D

데이터 불균형



Data size: n_k

Partial Loss: $F_k(w) = \frac{1}{n_k} \sum_{i \in D_k} f_i(w)$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

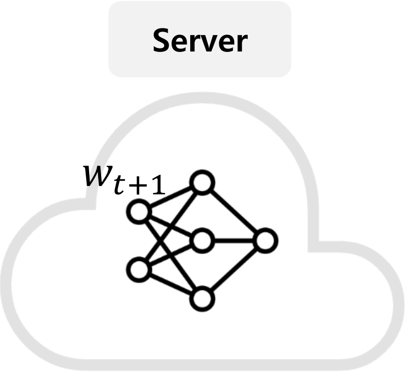
communication cost

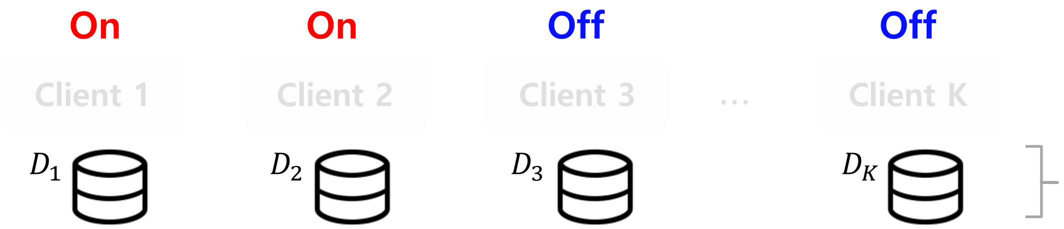
partial participation

Non-I.I.D

데이터 불균형

Goal | $w^* \triangleq \min_{w} f(w)$





Data size: n_k

Partial Loss: $F_k(w) = \frac{1}{n_k}\sum_{i \in D_k} f_i(w)$

매 시점마다, 전체 데이터 중 일부 데이터만을 가지고 학습?

Federated Averaging (FedAvg)

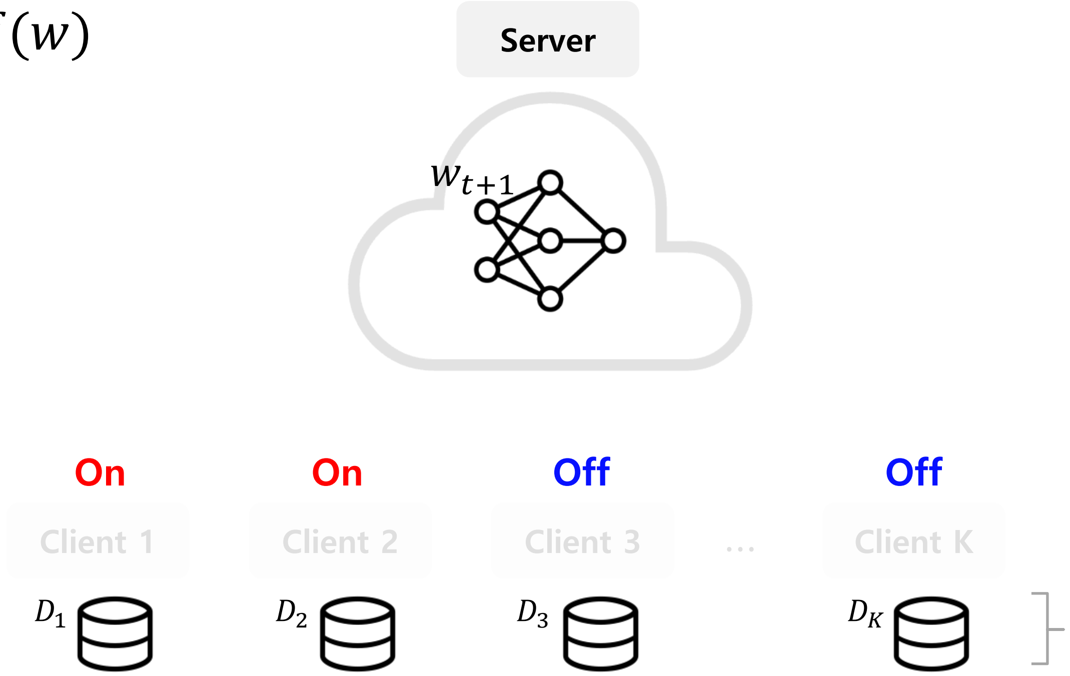
Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

- communication cost
- **partial participation**
- Non-I.I.D
- **데이터 불균형**

Goal | $w^* \triangleq \min_{w} f(w)$



Data size: n_k

Partial Loss: $F_k(w) = \frac{1}{n_k}\sum_{i \in D_k} f_i(w)$

매 시점마다, 전체 데이터 중 일부 데이터만을 가지고 학습?

Mini-Batch !!

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

communication cost

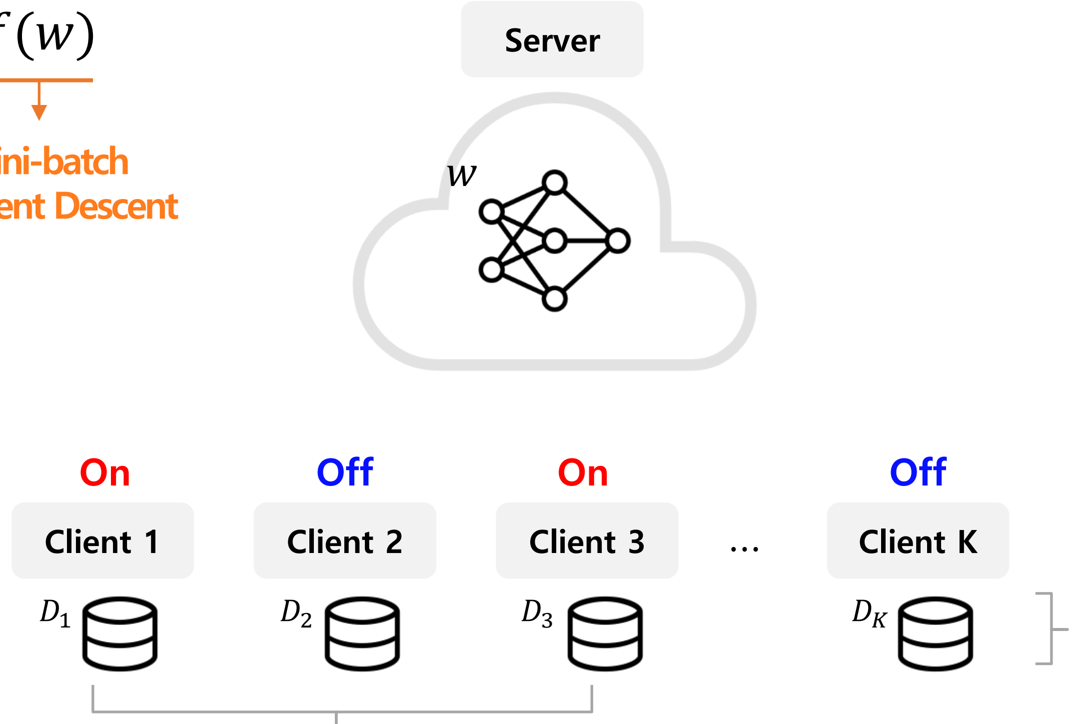
partial participation

Non-I.I.D

데이터 불균형

Goal | $w^* \triangleq \min_{w} f(w)$

$f(w)$ → Mini-batch Gradient Descent



Data size: n_k

Mini-batch size: $m = \sum_{k \in S_t} n_k$

Partial Loss: $F_k(w) = \frac{1}{n_k} \sum_{i \in D_k} f_i(w)$

$$f_B(w) = \frac{1}{m} \sum_{i} f_i(w) = \frac{1}{m} \sum_{k \in S_t} n_k F_k(w) = \sum_{k \in S_t} \frac{n_k}{m} F_k(w)$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

Goal | $w^* \triangleq \min_{w} f(w)$

Mini-batch Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f_B(w_t)$$

Mini-batch Loss Gradient

$$= w_t - \eta \nabla \{ \sum_{k \in S_t} \frac{n_k}{m} F_k(w_t) \}$$

$$= w_t - \sum_{k \in S_t} \frac{n_k}{m} \eta \nabla F_k(w_t)$$

Partial Loss Gradient

$$= \sum_{k \in S_t} \frac{n_k}{m} (w_t - \eta \nabla F_k(w_t))$$

Local Update

$$= \sum_{k \in S_t} \frac{n_k}{m} w_{t+1}^k$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

Goal | $w^* \triangleq \min_{w} f(w)$

Mini-batch Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f_B(w_t)$$

Mini-batch Loss Gradient

$$= w_t - \eta \nabla \{ \sum_{k \in S_t} \frac{n_k}{m} F_k(w_t) \}$$

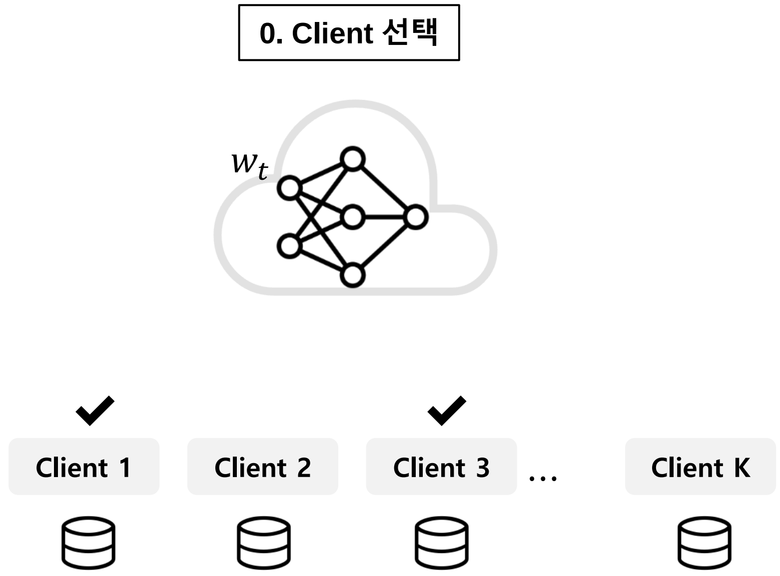
$$= w_t - \sum_{k \in S_t} \frac{n_k}{m} \eta \nabla F_k(w_t)$$

Partial Loss Gradient

$$= \sum_{k \in S_t} \frac{n_k}{m} (w_t - \eta \nabla F_k(w_t))$$

Local Update

$$= \sum_{k \in S_t} \frac{n_k}{m} w_{t+1}^k$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

Goal | $w^* \triangleq \min_{w} f(w)$

Mini-batch Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f_B(w_t)$$

Mini-batch Loss Gradient

$$= w_t - \eta \nabla \{ \sum_{k \in S_t} \frac{n_k}{m} F_k(w_t) \}$$

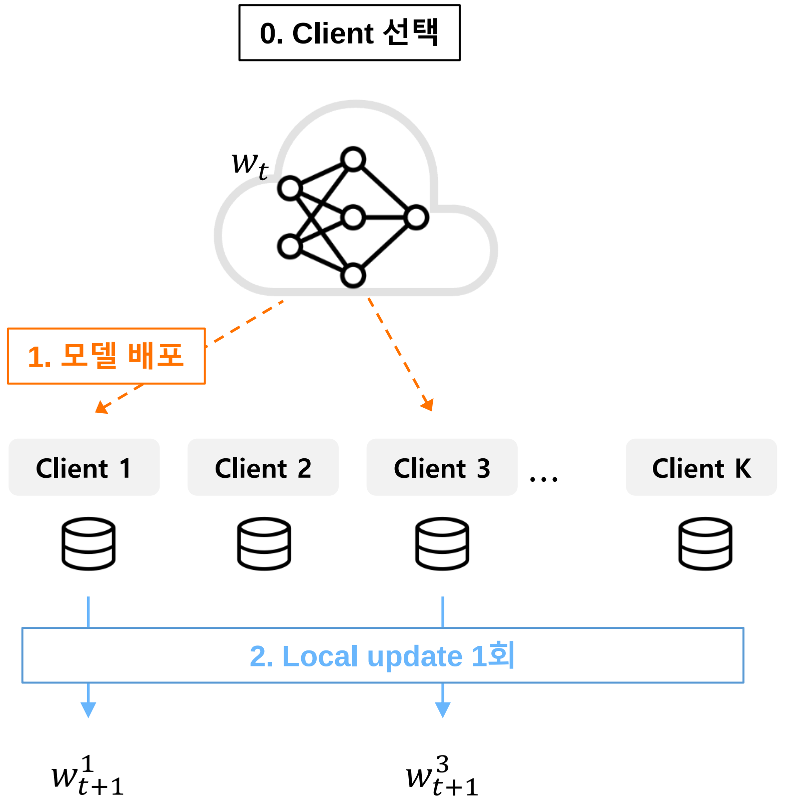
$$= w_t - \sum_{k \in S_t} \frac{n_k}{m} \eta \nabla F_k(w_t)$$

Partial Loss Gradient

$$= \sum_{k \in S_t} \frac{n_k}{m} (w_t - \eta \nabla F_k(w_t))$$

Local Update

$$= \sum_{k \in S_t} \frac{n_k}{m} w_{t+1}^k$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ FedSGD

➢ **Distributed Optimization에 partial participation과 데이터 불균형 반영**

Goal | $w^* \triangleq \min_{w} f(w)$

Mini-batch Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f_B(w_t)$$

Mini-batch Loss Gradient

$$= w_t - \eta \nabla \{ \sum_{k \in S_t} \frac{n_k}{m} F_k(w_t) \}$$

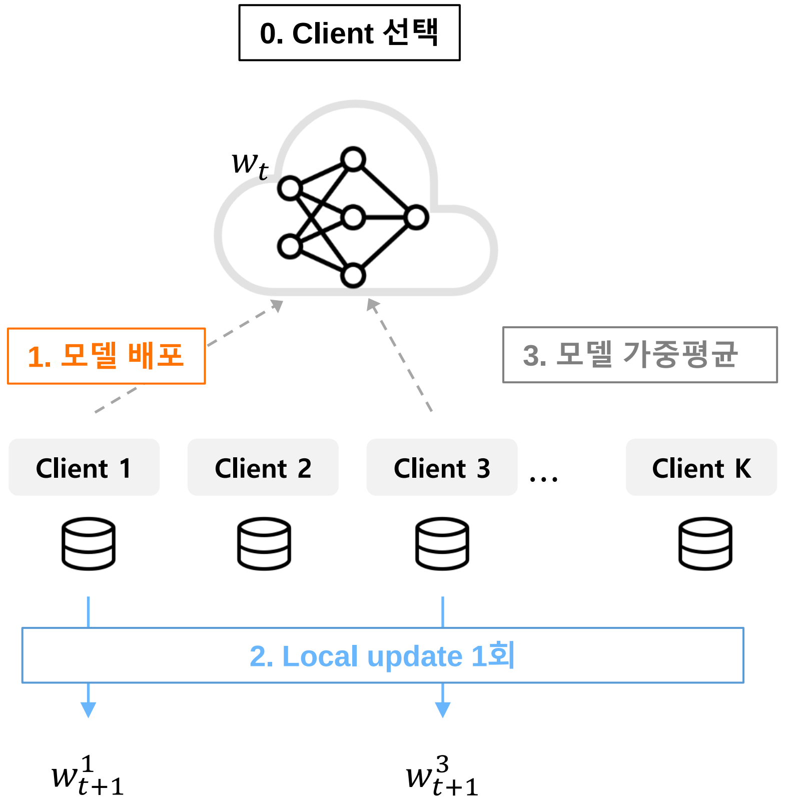
$$= w_t - \sum_{k \in S_t} \frac{n_k}{m} \eta \nabla F_k(w_t)$$

Partial Loss Gradient

$$= \sum_{k \in S_t} \frac{n_k}{m} (w_t - \eta \nabla F_k(w_t))$$

Local Update

$$= \sum_{k \in S_t} \frac{n_k}{m} w_{t+1}^k$$



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Federated Averaging (FedAvg)

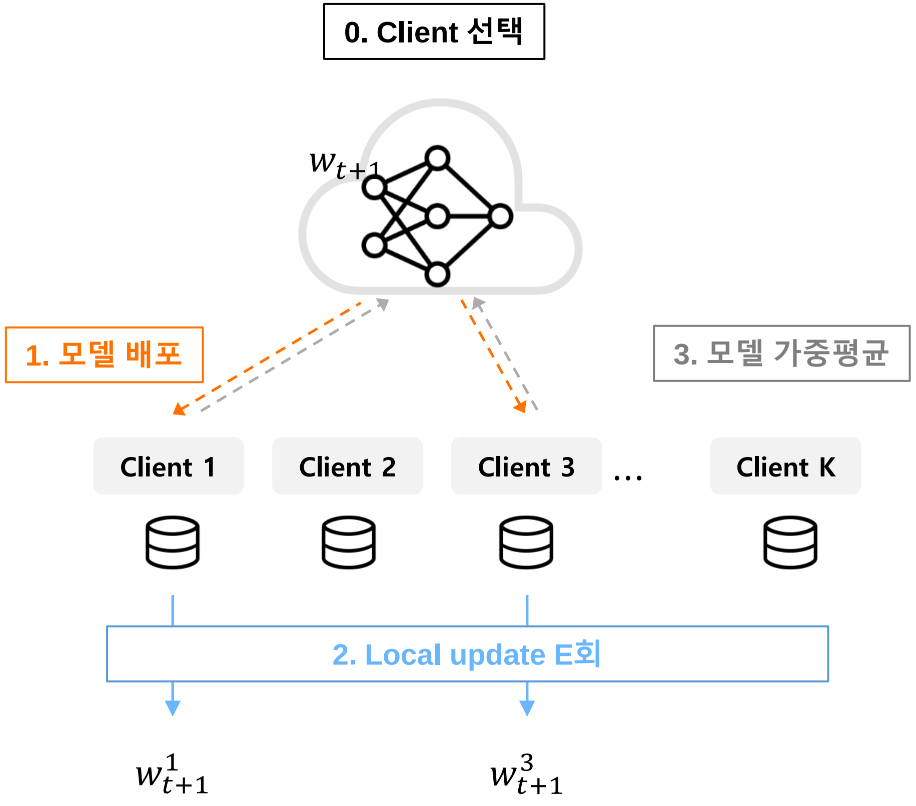
- **FedSGD:** Local update 당 communication 1회
- **FedAvg:** Local update E회당 communication 1회
- 동일 local update 수 기준, communication 횟수 $1/E$로 감소

communication cost

partial participation

Non-I.I.D

데이터 불균형



Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Federated Averaging (FedAvg)

- **FedSGD:** Local update 당 communication 1회
- **FedAvg:** Local update E 회당 communication 1회
- 동일 local update 수 기준, communication 횟수 $\mathbf{1/E}$로 감소

communication cost

partial participation

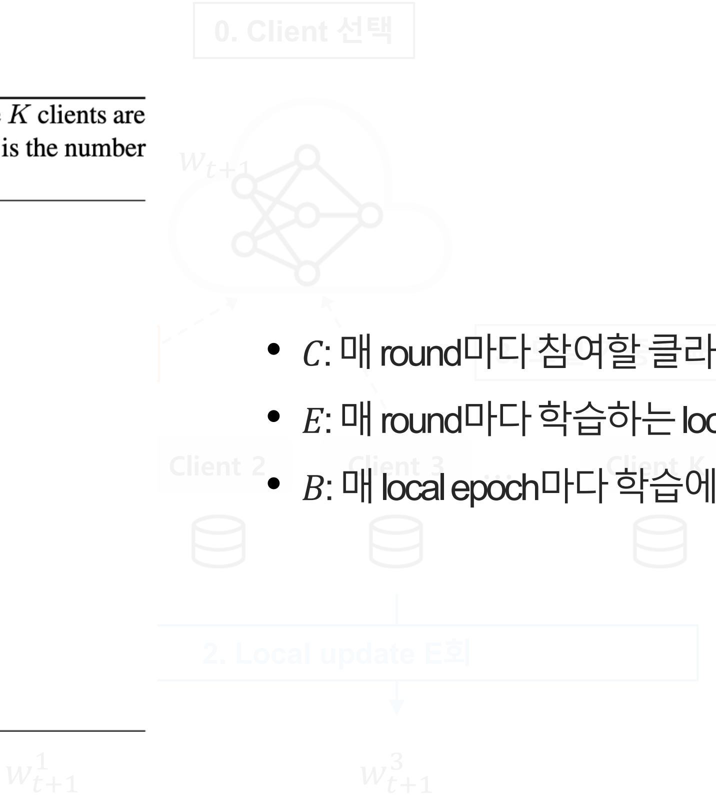
Non-I.I.D

데이터 불균형

Algorithm 1 FederatedAveraging. The K clients are indexed by k; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:
  initialize w_0
  for each round $t = 1, 2, \dots$ **do**
    $m \leftarrow \max(C \cdot K, 1)$
    $S_t \leftarrow$ (random set of m clients)
    for each client $k \in S_t$ **in parallel do**
      $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
    $m_t \leftarrow \sum_{k \in S_t} n_k$
    $w_{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{m_t} w_{t+1}^k$ // *Erratum*[4]

ClientUpdate(k, w): // *Run on client k*
  $\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)
  for each local epoch i from 1 to E **do**
    for batch $b \in \mathcal{B}$ **do**
      $w \leftarrow w - \eta \nabla \ell(w; b)$
  return w to server



- C: 매 round마다 참여할 클라이언트 비율 (Partial Participation)
- E: 매 round마다 학습하는 local epoch 수
- B: 매 local epoch마다 학습에 사용하는 local mini-batch 크기

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Federated Averaging (FedAvg)

- **FedSGD:** Local update 당 communication 1회
- **FedAvg:** Local update E회당 communication 1회
- 동일 local update 수 기준, communication 횟수 $1/E$로 감소

communication cost

partial participation

Non-I.I.D

데이터 불균형

1. Distributed Optimization이나
2. FedSGD와는 달리
'Local update 평균'이
'Global update'와 동일하다는 이론적 보장 X

Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t) \quad \text{Total Loss Gradient}$$
$$= w_t - \eta \nabla \{\frac{1}{K}\sum_{k=1}^{K} F_k(w)\}$$
$$= w_t - \frac{1}{K}\sum_{k=1}^{K} \eta \nabla F_k(w)$$
$$= \frac{1}{K}\sum_{k=1}^{K} (w_t - \eta \nabla F_k(w)) \quad \text{Partial Loss Gradient}$$
$$= \frac{1}{K}\sum_{k=1}^{K} w_{t+1}^k$$

Mini-batch Gradient Descent

$$w_{t+1} = w_t - \eta \nabla f(w_t) \quad \text{Mini-batch Loss Gradient}$$
$$= w_t - \eta \nabla \{\sum_{k \in S} \frac{n_k}{m} F_k(w)\}$$
$$= w_t - \sum_{k \in S} \frac{n_k}{m} \eta \nabla F_k(w)$$
$$= \sum_{k \in S} \frac{n_k}{m} (w_t - \eta \nabla F_k(w)) \quad \text{Partial Loss Gradient}$$
$$= \sum_{k \in S} \frac{n_k}{m} w_{t+1}^k$$

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Federated Averaging (FedAvg)

- **FedSGD:** Local update 당 communication 1회
- **FedAvg:** Local update $\boldsymbol{E}$회당 communication 1회
- 동일 local update 수 기준, communication 횟수 $\mathbf{1/\boldsymbol{E}}$로 감소

communication cost

partial participation

Non-I.I.D

데이터 불균형

1. Distributed Optimization이나
2. FedSGD와는 달리
‘Local update 평균’이
‘Global update’와 동일하다는 이론적 보장 X

따라서, FedAvg에 대한 수렴성 증명 필요!
On the Convergence of FedAvg on Non-IID Data (ICLR’20)

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Experiments

➢ 하이퍼 파라미터 *C*, *E*, *B*를 어떻게 설정해야 하는가?

- *C*: 매 round마다 참여할 클라이언트 비율 (Partial Participation)
- *E*: 매 round마다 학습하는 local epoch 수
- *B*: 매 local epoch마다 학습에 사용하는 local mini-batch 크기

2NN	IID		Non-IID	
C	$B = \infty$	$B = 10$	$B = \infty$	$B = 10$
0.0	1455	316	4278	3275
0.1	1474 (1.0×)	87 (3.6×)	1796 (2.4×)	664 (4.9×)
0.2	1658 (0.9×)	77 (4.1×)	1528 (2.8×)	619 (5.3×)
0.5	— (—)	75 (4.2×)	— (—)	443 (7.4×)
1.0	— (—)	70 (4.5×)	— (—)	380 (8.6×)
CNN, $E = 5$				
0.0	387	50	1181	956
0.1	339 (1.1×)	18 (2.8×)	1100 (1.1×)	206 (4.6×)
0.2	337 (1.1×)	18 (2.8×)	978 (1.2×)	200 (4.8×)
0.5	164 (2.4×)	18 (2.8×)	1067 (1.1×)	261 (3.7×)
1.0	246 (1.6×)	16 (3.1×)	— (—)	97 (9.9×)

Test accuracy 97% 도달 round (2NN)

Test accuracy 99% 도달 round (CNN)

클라이언트 당 label 2개씩만 할당 (Label Shift)

MNIST 데이터셋 분류 실험

Federated Averaging (FedAvg)

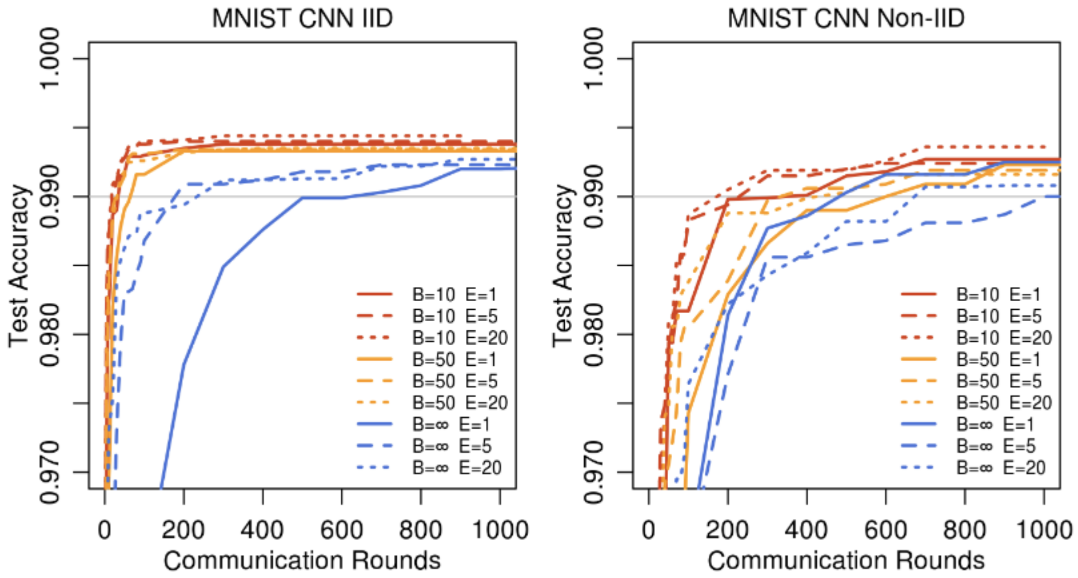
Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Experiments

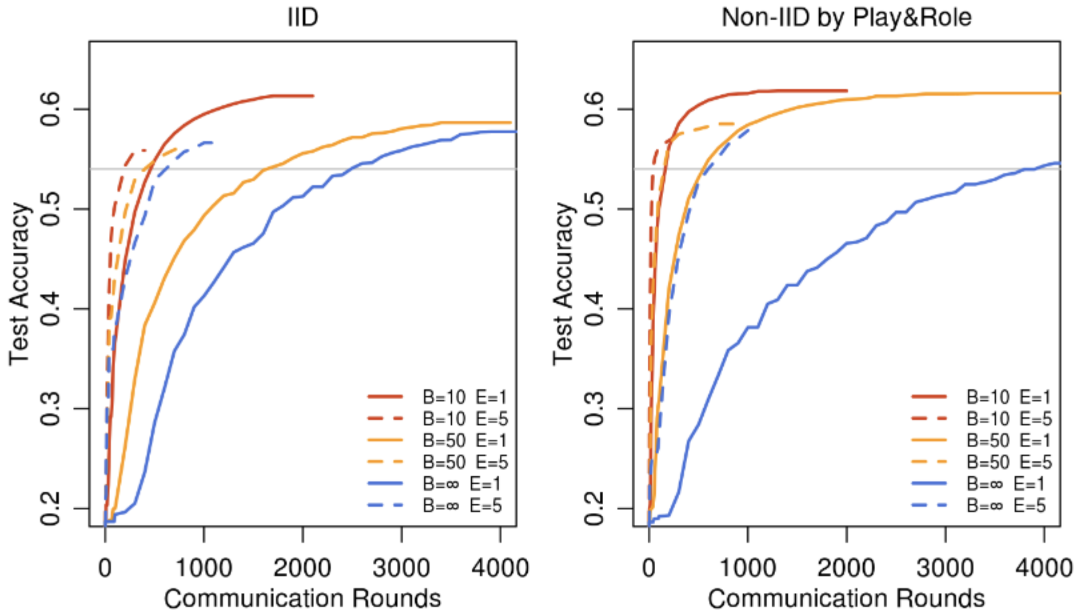
➢ 하이퍼 파라미터 C, E, B를 어떻게 설정해야 하는가?

- C: 매 round마다 참여할 클라이언트 비율 (Partial Participation)
- E: 매 round마다 학습하는 local epoch 수
- B: 매 local epoch마다 학습에 사용하는 local mini-batch 크기

$$\boldsymbol{Local\ Computation} \propto \frac{\boldsymbol{E}}{\boldsymbol{B}}$$



MNIST 데이터셋 분류 실험



Shakespeare Next-token prediction 실험

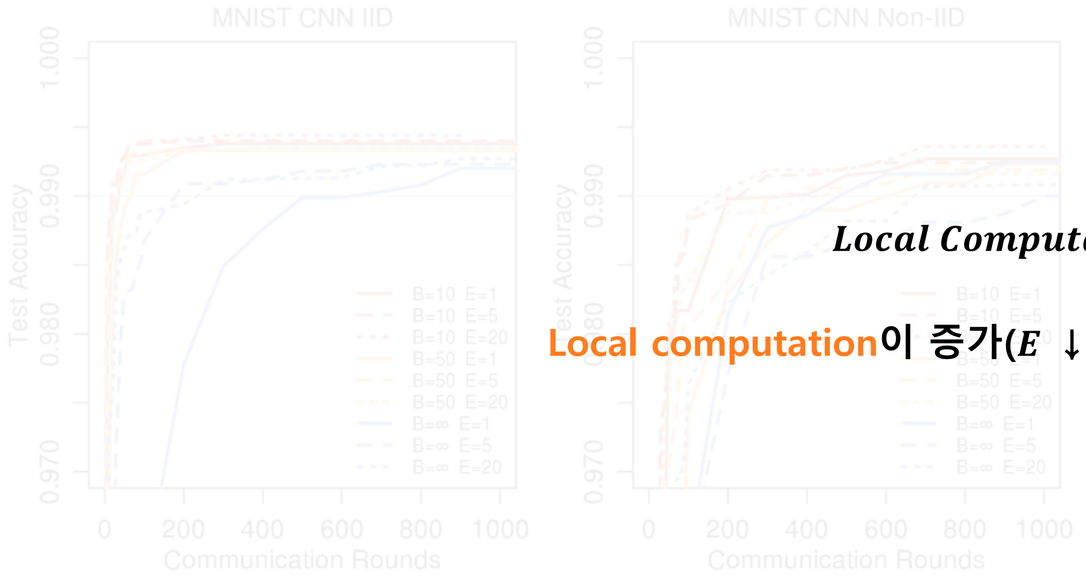
Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

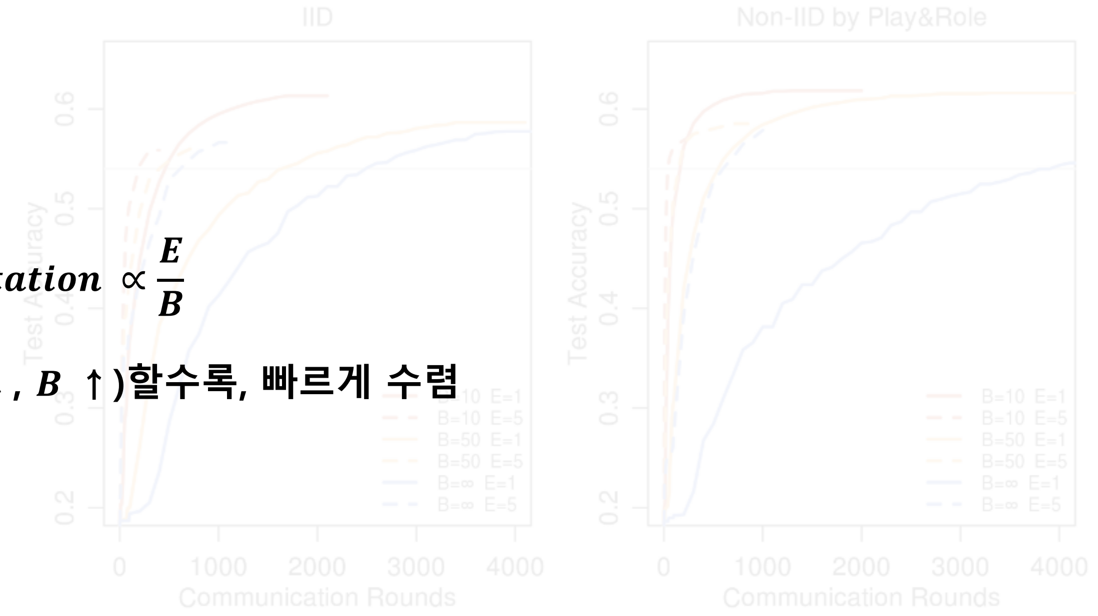
❖ Experiments

➢ 하이퍼 파라미터 C, E, B를 어떻게 설정해야 하는가?

- C: 매 round마다 참여할 클라이언트 비율 (Partial Participation)
- E: 매 round마다 학습하는 local epoch 수
- B: 매 local epoch마다 학습에 사용하는 local mini-batch 크기



MNIST 데이터셋 분류 실험



Shakespeare Next-token prediction 실험

$$Local\ Computation \propto \frac{E}{B}$$

Local computation이 증가(E ↓, B ↑)할수록, 빠르게 수렴

Federated Averaging (FedAvg)

Communication-Efficient Learning of Deep Networks from Decentralized Data

❖ Discussion

➢ 의의: Federated Learning과 이를 위한 FedAvg 알고리즘 제안

➢ 한계점:

- 개선의 여지가 많은 알고리즘 (e.g. Non-IID 조건에 대한 고려 X) ➡ Federated Optimization in Heterogeneous Networks (MLSys'20)
- 실험적으로 잘 작동함을 보였으나, 엄밀한 설명 부족 ➡ On the Convergence of FedAvg on Non-IID Data (ICLR'20)

FedProx

Federated Optimization in Heterogeneous Networks

❖ **FedAvg 한계점 개선한 알고리즘 제안 및 convergence analysis 제공**

- ➢ MLSys'20
- ➢ 피인용 6028회 (2025년 3월 기준)

FEDERATED OPTIMIZATION IN HETEROGENEOUS NETWORKS

Tian Li[1] Anit Kumar Sahu[2] Manzil Zaheer[3] Maziar Sanjabi[4] Ameet Talwalkar[1,5] Virginia Smith[1]

ABSTRACT

Federated Learning is a distributed learning paradigm with two key challenges that differentiate it from traditional distributed optimization: (1) significant variability in terms of the systems characteristics on each device in the network (systems heterogeneity), and (2) non-identically distributed data across the network (statistical heterogeneity). In this work, we introduce a framework, FedProx, to tackle heterogeneity in federated networks. FedProx can be viewed as a generalization and re-parametrization of FedAvg, the current state-of-the-art method for federated learning. While this re-parameterization makes only minor modifications to the method itself, these modifications have important ramifications both in theory and in practice. Theoretically, we provide convergence guarantees for our framework when learning over data from non-identical distributions (statistical heterogeneity), and while adhering to device-level systems constraints by allowing each participating device to perform a variable amount of work (systems heterogeneity). Practically, we demonstrate that FedProx allows for more robust convergence than FedAvg across a suite of realistic federated datasets. In particular, in highly heterogeneous settings, FedProx demonstrates significantly more stable and accurate convergence behavior relative to FedAvg—improving absolute test accuracy by 22% on average.

1 INTRODUCTION

Federated learning has emerged as an attractive paradigm for distributing training of machine learning models in networks of remote devices. While there is a wealth of work on distributed optimization in the context of machine learning, two key challenges distinguish federated learning from traditional distributed optimization: high degrees of *systems and statistical heterogeneity*[1] (McMahan et al., 2017; Li et al., 2019).

In an attempt to handle heterogeneity and tackle high communication costs, optimization methods that allow for local updating and low participation are a popular approach for federated learning (McMahan et al., 2017; Smith et al., 2017). In particular, FedAvg (McMahan et al., 2017) is an iterative method that has emerged as the de facto optimization method in the federated setting. At each iteration, FedAvg first locally performs E epochs of stochastic gradient descent (SGD) on K devices—where E is a small constant and K is a small fraction of the total devices in the network. The devices then communicate their model updates to a central server, where they are averaged.

While FedAvg has demonstrated empirical success in heterogeneous settings, it does not fully address the underlying challenges associated with heterogeneity. In the context of systems heterogeneity, FedAvg does not allow participating devices to perform variable amounts of local work based on their underlying systems constraints; instead it is common to simply drop devices that fail to compute E epochs within a specified time window (Bonawitz et al., 2019). From a statistical perspective, FedAvg has been shown to diverge empirically in settings where the data is non-identically distributed across devices (e.g., McMahan et al., 2017, Sec 3). Unfortunately, FedAvg is difficult to analyze theoretically in such realistic scenarios and thus lacks convergence guarantees to characterize its behavior

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems(MLSys)*, 2, 429–450.

FedProx

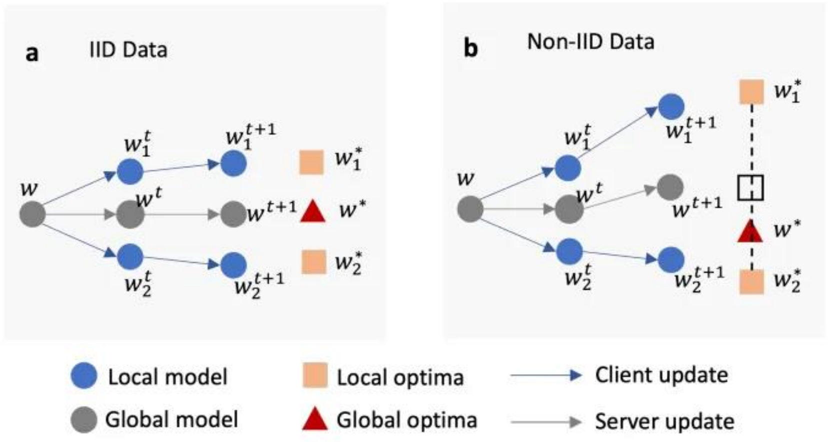
Federated Optimization in Heterogeneous Networks

❖ FedAvg 한계점

➢ 클라이언트 간 이질성에 취약

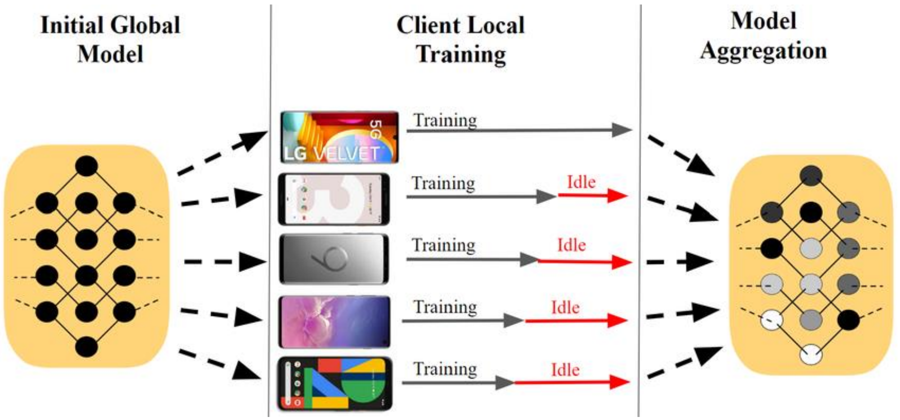
- **통계적 이질성(Non-IID)**: 데이터 분포 차이가 클 경우, 수렴하지 못하고 발산 (Client Drift)
- **시스템적 이질성**: 학습 속도가 느린 디바이스 존재 (Straggler) → 동일 Epoch 강제는 비효율적

[통계적 이질성]



Local 학습 시 Global model에서 지나치게 멀어지는 것을 방지

[시스템적 이질성]



디바이스 사양에 따라 Epoch 수 조절

[1] https://ar5iv.labs.arxiv.org/html/2103.00710
[2] https://www.researchgate.net/figure/Stragglers-impact-on-FL-performance-In-synchronous-FL-all-clients-wait-for-the_fig1_372163244

FedProx

Federated Optimization in Heterogeneous Networks

❖ Proximal Term & γ_t^k-inexact Solution

➢ 두 이질성에 대한 보완책

- **Proximal Term**: 로컬 모델과 글로벌 모델 간 차이를 억제 → 통계적 이질성 ↓
- **γ_t^k-inexact Solution** : 학습 속도가 느린 디바이스는 다소 부정확한 모델이라도 반환할 수 있도록 허락 → 시스템적 이질성 ↓

[Proximal Term]

$$\min_w h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$$

Proximal Term

Local 학습 시 Global model에서
지나치게 멀어지는 것을 방지

[γ_t^k-inexact Solution]

디바이스 사양에 따라 Epoch 수 조절

FedProx

Federated Optimization in Heterogeneous Networks

❖ Proximal Term & γ_t^k-inexact Solution

➢ 두 이질성에 대한 보완책

- Proximal Term: 로컬 모델과 글로벌 모델 간 차이를 억제 → 통계적 이질성 ↓
- γ_t^k-inexact Solution : 학습 속도가 느린 디바이스는 다소 부정확한 모델이라도 반환할 수 있도록 허락 → 시스템적 이질성 ↓

[Proximal Term]

$$\min_w h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$$

Proximal Term

$$\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$$

페널티 파라미터

Local 학습 시 Global model에서
지나치게 멀어지는 것을 방지

[γ_t^k-inexact Solution]

디바이스 사양에 따라 Epoch 수 조절

FedProx

Federated Optimization in Heterogeneous Networks

❖ Proximal Term & γ_t^k-inexact Solution

➢ 두 이질성에 대한 보완책

- Proximal Term: 로컬 모델과 글로벌 모델 간 차이를 억제 → 통계적 이질성 ↓
- γ_t^k-inexact Solution : 학습 속도가 느린 디바이스는 다소 부정확한 모델이라도 반환할 수 있도록 허락 → 시스템적 이질성 ↓

[Proximal Term]

$$\min_w h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$$

Proximal Term: $\frac{\mu}{2}\|w - w_t\|^2$

$$\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$$

페널티 파라미터: μ

Local 학습 시 Global model에서
지나치게 멀어지는 것을 방지

[γ_t^k-inexact Solution]

$$\|\nabla h_k(w^*; w_t)\| \leq \gamma_t^k \|\nabla h_k(w_t; w_t)\|$$

$$\|\nabla F_k(w^*) + \mu(w^* - w_t)\| \leq \gamma_t^k \|\nabla F_k(w_t)\|$$

디바이스 사양에 따라 Epoch 수 조절

FedProx

Federated Optimization in Heterogeneous Networks

❖ Proximal Term & γ_t^k-inexact Solution

➢ 두 이질성에 대한 보완책

- Proximal Term: 로컬 모델과 글로벌 모델 간 차이를 억제 → 통계적 이질성 ↓
- γ_t^k-inexact Solution : 학습 속도가 느린 디바이스는 다소 부정확한 모델이라도 반환할 수 있도록 허락 → 시스템적 이질성 ↓

[Proximal Term]

$$\min_w h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$$

Proximal Term

$$\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$$

페널티 파라미터

Local 학습 시 Global model에서
지나치게 멀어지는 것을 방지

[γ_t^k-inexact Solution]

$$\|\nabla h_k(w^*; w_t)\| \leq \gamma_t^k \|\nabla h_k(w_t; w_t)\|$$

$$\|\nabla F_k(w^*) + \mu(w^* - w_t)\| \leq \gamma_t^k \|\nabla F_k(w_t)\|$$

부정확한 로컬 모델이 글로벌
모델에 악영향을 주지는 않을까?

디바이스 사양에 따라 Epoch 수 조절

FedProx

Federated Optimization in Heterogeneous Networks

❖ Proximal Term & γ_t^k-inexact Solution

➢ 두 이질성에 대한 보완책

- Proximal Term: 로컬 모델과 글로벌 모델 간 차이를 억제 → 통계적 이질성 ↓
- γ_t^k-inexact Solution : 학습 속도가 느린 디바이스는 다소 부정확한 모델이라도 반환할 수 있도록 허락 → 시스템적 이질성 ↓

[Proximal Term]

$$\min_w h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$$

Proximal Term

$$\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$$

페널티 파라미터

Local 학습 시 Global model에서
지나치게 멀어지는 것을 방지

[γ_t^k-inexact Solution]

$$\|\nabla h_k(w^*; w_t)\| \le \gamma_t^k \|\nabla h_k(w_t; w_t)\|$$

$$\|\nabla F_k(w^*) + \mu(w^* - w_t)\| \le \gamma_t^k \|\nabla F_k(w_t)\|$$

부정확한 로컬 모델이 글로벌
모델에 악영향을 주지는 않을까?

디바이스 사양에 따라 Epoch 수 조절

FedProx

Federated Optimization in Heterogeneous Networks

❖ FedProx 알고리즘

➢ FedAvg 알고리즘에서 E (local train epoch 수) 대체 → μ, γ

Algorithm 1 Federated Averaging (FedAvg)

Input: $K, T, \eta, E, w^0, N, p_k, k = 1, \cdots, N$

for $t = 0, \cdots, T-1$ **do**

Server selects a subset S_t of K devices at random (each device k is chosen with probability p_k)

Server sends w^t to all chosen devices

Each device $k \in S_t$ updates w^t for E epochs of SGD on F_k with step-size η to obtain w_k^{t+1}

Each device $k \in S_t$ sends w_k^{t+1} back to the server

Server aggregates the w's as $w^{t+1} = \frac{1}{K}\sum_{k \in S_t} w_k^{t+1}$

end for

Algorithm 2 FedProx (Proposed Framework)

Input: $K, T, \mu, \gamma, w^0, N, p_k, k = 1, \cdots, N$

for $t = 0, \cdots, T-1$ **do**

Server selects a subset S_t of K devices at random (each device k is chosen with probability p_k)

Server sends w^t to all chosen devices

Each chosen device $k \in S_t$ finds a w_k^{t+1} which is a γ_k^t-inexact minimizer of: $w_k^{t+1} \approx \arg\min_w h_k(w; w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2$

Each device $k \in S_t$ sends w_k^{t+1} back to the server

Server aggregates the w's as $w^{t+1} = \frac{1}{K}\sum_{k \in S_t} w_k^{t+1}$

end for

FedProx

Federated Optimization in Heterogeneous Networks

❖ Convergence Analysis

➢ **FedProx 사용 시, total Loss $f(w)$가 최솟값 f^*으로 수렴함을 증명**

Definition. B-local dissimilarity

$$\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right] = \|\nabla f(w)\|^2 B(w)^2$$

Data Mining
Quality Analytics

FedProx

Federated Optimization in Heterogeneous Networks

❖ Convergence Analysis

➢ FedProx 사용 시, total Loss $f(w)$가 최솟값 f^*으로 수렴함을 증명

Definition. B-local dissimilarity

$$\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right] = \|\nabla f(w)\|^2 B(w)^2$$

$$\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right] = \|\mathbb{E}_k\left[\nabla F_k(w)\right]\|^2 B(w)^2$$

분산과 유사!

$\nabla F_1(w) = 1 \quad \nabla F_2(w) = 1 \quad \nabla F_3(w) = 1$

$$1 = B(w)^2$$

$\nabla F_1(w) = -3 \quad \nabla F_2(w) = 3 \quad \nabla F_3(w) = 6$

$$1.5 = B(w)^2$$

FedProx

Federated Optimization in Heterogeneous Networks

❖ Convergence Analysis

➢ **FedProx 사용 시, total Loss $f(w)$가 최솟값 f^*으로 수렴함을 증명**

Definition. B-local dissimilarity

$$\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right] = \|\nabla f(w)\|^2 B(w)^2$$

Assumption. Bounded dissimilarity

$$B(w) \leq B$$

FedProx

Federated Optimization in Heterogeneous Networks

❖ Convergence Analysis

➢ **FedProx 사용 시, total Loss $f(w)$가 최솟값 f^*으로 수렴함을 증명**

Definition. B-local dissimilarity

$$\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right] = \|\nabla f(w)\|^2 B(w)^2$$

Assumption. Bounded dissimilarity

$$B(w) \leq B$$

Theorem 1. Non-convex FedProx convergence

$$\mathbb{E}_{S_t}\left[f(w_{t+1})\right] \leq f(w_t) - \rho\|\nabla f(w_t)\|^2$$

$B,\ \mu,\ \gamma$에 대한 함수
$\rho > 0$일 때, 수렴

Theorem 2. Convergence rate

$$f(w_0) - f^* =: \Delta$$

$$T := O\left(\frac{\Delta}{\rho\epsilon}\right)$$

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla f(w_t)\|^2\right] \leq \epsilon$$

FedProx

Federated Optimization in Heterogeneous Networks

❖ Convergence Analysis

➢ **FedProx 사용 시, total Loss $f(w)$가 최솟값 f^*으로 수렴함을 증명**

Definition. B-local dissimilarity

$$\mathbb{E}_k \left[\|\nabla F_k(w)\|^2 \right] = \|\nabla f(w)\|^2 B(w)^2$$

Assumption. Bounded dissimilarity

$$B(w) \leq B$$

Theorem 1. Non-convex FedProx convergence

$$\mathbb{E}_{S_t} \left[f(w_{t+1}) \right] \leq f(w_t) - \rho \|\nabla f(w_t)\|^2$$

Theorem 2. Convergence rate

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left[\|\nabla f(w_t)\|^2 \right] \leq \epsilon$$

$$f(w_0) - f^* =: \Delta$$

$$T := O\left(\frac{\Delta}{\rho\epsilon} \right)$$

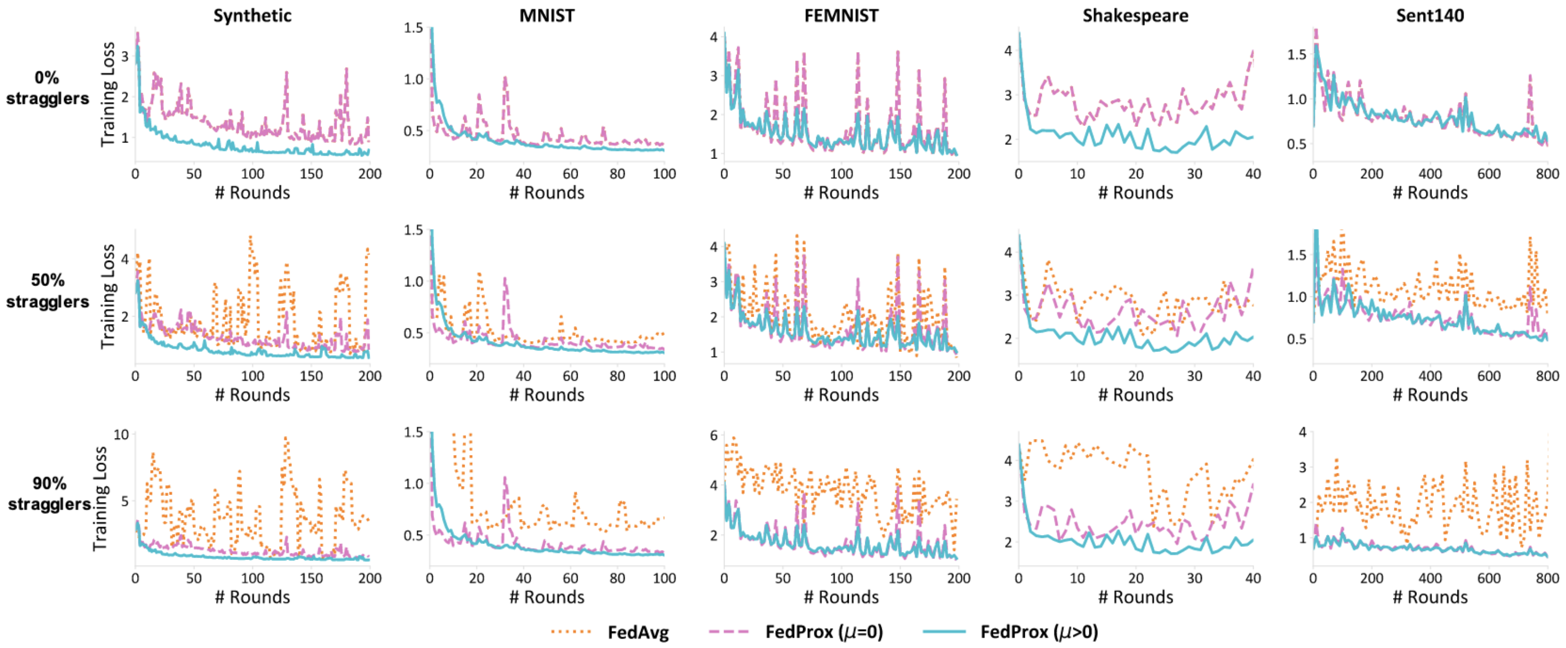
FedProx

Federated Optimization in Heterogeneous Networks

❖ Experiments

➢ 시스템적 이질성

- 매 round마다 0%, 50%, 혹은 90%의 straggler가 발생
- FedAvg는 Straggler를 FL에서 배제하도록 설정



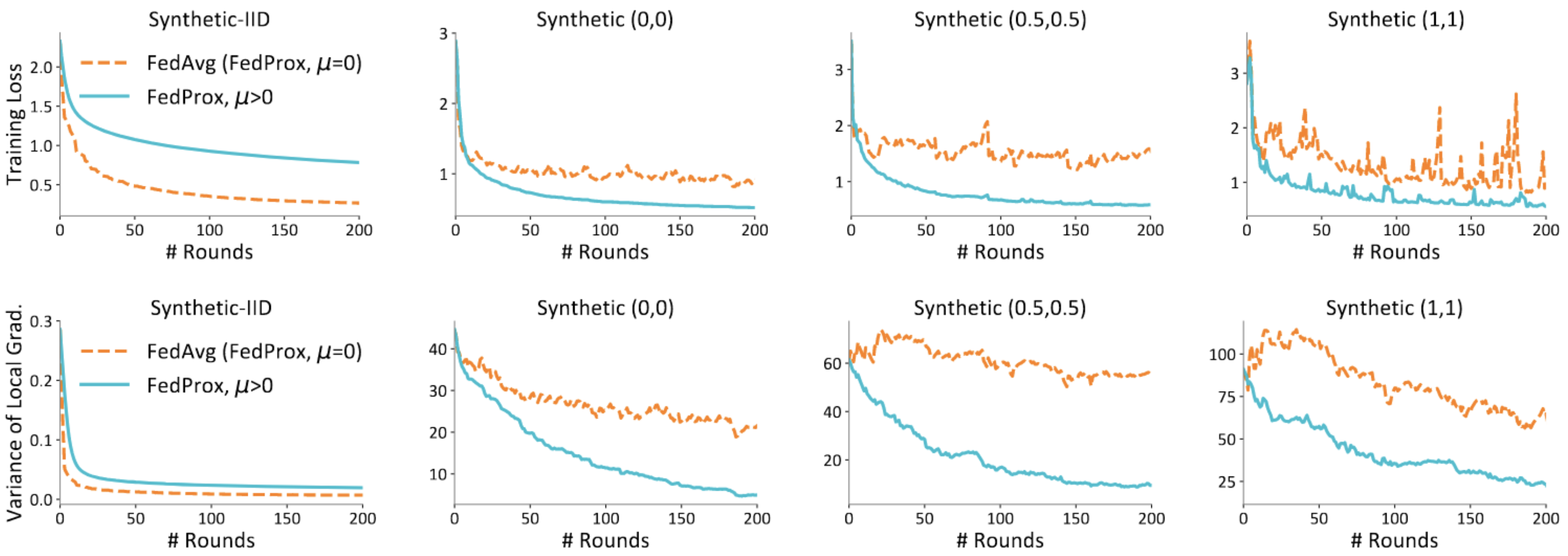
FedProx

Federated Optimization in Heterogeneous Networks

❖ Experiments

➢ 통계적 이질성

- Straggler는 존재하지 않는다고 가정
- 클라이언트 간 통계적 이질성을 달리 하며 실험 진행



Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ **FedAvg에 대한 convergence analysis 제공**

- ICLR'20
- 피인용 2832회 (2025년 3월 기준)

Published as a conference paper at ICLR 2020

ON THE CONVERGENCE OF FEDAVG ON NON-IID DATA

Xiang Li*
School of Mathematical Sciences
Peking University
Beijing, 100871, China
smslixiang@pku.edu.cn

Kaixuan Huang*
School of Mathematical Sciences
Peking University
Beijing, 100871, China
hackyhuang@pku.edu.cn

Wenhao Yang*
Center for Data Science
Peking University
Beijing, 100871, China
yangwenhaosms@pku.edu.cn

Shusen Wang
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ 07030, USA
shusen.wang@stevens.edu

Zhihua Zhang
School of Mathematical Sciences
Peking University
Beijing, 100871, China
zhzhang@math.pku.edu.cn

ABSTRACT

Federated learning enables a large amount of edge computing devices to jointly learn a model without data sharing. As a leading algorithm in this setting, Federated Averaging (FedAvg) runs Stochastic Gradient Descent (SGD) in parallel on a small subset of the total devices and averages the sequences only once in a while. Despite its simplicity, it lacks theoretical guarantees under realistic settings. In this paper, we analyze the convergence of FedAvg on non-iid data and establish a convergence rate of $\mathcal{O}(\frac{1}{T})$ for strongly convex and smooth problems, where T is the number of SGDs. Importantly, our bound demonstrates a trade-off between communication-efficiency and convergence rate. As user devices may be disconnected from the server, we relax the assumption of full device participation to partial device participation and study different averaging schemes; low device participation rate can be achieved without severely slowing down the learning. Our results indicates that heterogeneity of data slows down the convergence, which matches empirical observations. Furthermore, we provide a necessary condition for FedAvg on non-iid data: the learning rate η must decay, even if full-gradient is used; otherwise, the solution will be $\Omega(\eta)$ away from the optimal.

Li, X., Huang, K., Yang, W., Wang, S., & Zhang, Z. (2020). On the convergence of FedAvg on non-IID data. *International Conference on Learning Representations (ICLR).*

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ **증명 과정**

- Full participation에 대해 증명 후, partial participation으로 확장
- Partial participation의 sampling과 averaging 전략은 아래와 같음

2023년 개정 전 알고리즘

Client selection

	Paper	Sampling	Averaging	Convergence rate
FedAvg 제안 논문	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$
	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$

Sampling and Averaging Schemes

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 증명 과정

- Full participation에 대해 증명 후, partial participation으로 확장
- Partial participation의 sampling과 averaging 전략은 아래와 같음

	개정 전	개정 후
FedAvg 제안 논문	$w_{t+1} = \sum_{k \notin S_t} \frac{n_k}{n} w_t + \sum_{k \in S_t} \frac{n_k}{n} w_{t+1}^k$	$w_{t+1} = \sum_{k \in S_t} \frac{n_k}{m_t} w_{t+1}^k$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 증명 과정

- ➢ Full participation에 대해 증명 후, partial participation으로 확장
- ➢ Partial participation의 sampling과 averaging 전략은 아래와 같음

2023년 개정 전 알고리즘

Client selection

	Paper	Sampling	Averaging	Convergence rate	
FedAvg 제안 논문	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-	**수렴성 보장 실패**
	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$	
	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$	

Sampling and Averaging Schemes

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 증명 과정

- Full participation에 대해 증명 후, partial participation으로 확장
- Partial participation의 sampling과 averaging 전략은 아래와 같음

Client selection

	Paper	Sampling	Averaging	Convergence rate
	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
FedProx 제안 논문	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[5]
	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[6]

Sampling and Averaging Schemes

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 증명 과정

- Full participation에 대해 증명 후, partial participation으로 확장
- Partial participation의 sampling과 averaging 전략은 아래와 같음

Client selection

Paper	Sampling	Averaging	Convergence rate
McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$
Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$

수렴성 보장을 위한 변형

Sampling and Averaging Schemes

$$\sum_{k \in S_t} \frac{n_k}{n} \neq 1$$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 증명 과정

- ➢ Full participation에 대해 증명 후, partial participation으로 확장
- ➢ Partial participation의 sampling과 averaging 전략은 아래와 같음

Client selection

수렴성 보장을 위한 변형

Paper	Sampling	Averaging	Convergence rate
McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$
Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$

Sampling and Averaging Schemes

$$\mathbb{E}\left[\frac{N}{K} \sum_{k \in \mathcal{S}_t} \frac{n_k}{n}\right] = 1$$

N: 전체 클라이언트 수

K: 한 라운드에 클라이언트 수 제한

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ 가설

- FedProx와는 달리, convexity에 대한 가정 존재
- DNN 적용에 한계 (Non-convex)

Assumptions

Assumption 1: L-smoothness

$$F_k(v) - F_k(w) \leq (v - w)^T \nabla F_k(w) + \frac{L}{2}\|v - w\|_2^2.$$

Assumption 2: μ-strong convexity

$$F_k(v) - F_k(w) \geq (v - w)^T \nabla F_k(w) + \frac{\mu}{2}\|v - w\|_2^2.$$

Assumption 3: Bounded Variance

$$\mathbb{E}[\|\nabla F_k(w_k^t, \xi_k^t) - \nabla F_k(w_k^t)\|^2] \leq \sigma_k^2.$$

Assumption 4: Uniformly Bounded Squared Expectation

$$\mathbb{E}[\|\nabla F_k(w_k^t, \xi_k^t)\|^2] \leq G^2.$$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Full Participation

- 최종 시점 모델에 대한 손실 값이 이론적 최솟값에 수렴함을 증명
- T: 전체 local update 수
- E: Communication 1회 당 local update 수

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}) \left(\frac{2B}{\mu} + \frac{\mu\gamma}{2} \mathbb{E}\|w_1 - w^*\|^2 \right),$$

$$B = \sum_{k=1}^{N} p_k^2 \sigma_k^2 + 6L\Gamma + 8(E-1)^2 G^2$$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Full Participation

- 최종 시점 모델에 대한 손실 값이 이론적 최솟값에 수렴함을 증명
- T: 전체 local update 수
- E: Communication 1회 당 local update 수

T가 증가함에 따라 0으로 수렴

이론적 최솟값

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2B}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|w_1 - w^*\|^2\right),$$

T 시점 모델 손실 값

$$B = \sum_{k=1}^{N} p_k^2 \sigma_k^2 + 6L\Gamma + 8(E-1)^2 G^2$$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Partial Participation

- ➢ 두 가지 전략에 대해서 증명
- ➢ Scheme 2에 대해서는, **데이터 균형**을 가정 (비현실적 가정)

	Paper	Sampling	Averaging	Convergence rate
	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Scheme 1	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[5]
Scheme 2	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[6]

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B + C)}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|w_1 - w^*\|^2\right),$$

Scheme 1

$$C = \frac{4}{K} E^2 G^2$$

Scheme 2 + Balanced Data Assumption

$$C = \frac{N - K}{N - 1} \frac{4}{K} E^2 G^2$$

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Partial Participation

- 두 가지 전략에 대해서 증명
- Scheme 2에 대해서는, **데이터 균형**을 가정 (비현실적 가정)

	Paper	Sampling	Averaging	Convergence rate
	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Scheme 1	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[5]
Scheme 2	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[6]

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}) \left(\frac{2(B + C)}{\mu} + \frac{\mu\gamma}{2} \mathbb{E}\|w_1 - w^*\|^2 \right),$$

Scheme 1 – N개의 디바이스 중 K개를 임의로 Sampling 할 수 있어야 함

샘플링 된 디바이스 중 straggler가 존재한다면 비효율적

Data Mining Quality Analytics

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Partial Participation

- ➢ 두 가지 전략에 대해서 증명
- ➢ Scheme 2에 대해서는, **데이터 균형**을 가정 (비현실적 가정)

	Paper	Sampling	Averaging	Convergence rate
Scheme 1	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin \mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Scheme 2	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$
	Ours	$\mathcal{S}_t \sim \mathcal{U}(N, K)$	$\sum_{k \in \mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B + C)}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|w_1 - w^*\|^2\right),$$

Scheme 2 **– N개의 디바이스 중 먼저 학습이 끝난 K개를 선택**

Straggler가 존재한다면 *Scheme 1*의 현실적 대안이 될 수 있음

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Partial Participation

- ➢ 두 가지 전략에 대해서 증명
- ➢ Scheme 2에 대해서는, **데이터 균형**을 가정 (비현실적 가정)

	Paper	Sampling	Averaging	Convergence rate
	McMahan et al. (2017)	$\mathcal{S}_t \sim \mathcal{U}(N,K)$	$\sum_{k\notin\mathcal{S}_t} p_k \mathbf{w}_t + \sum_{k\in\mathcal{S}_t} p_k \mathbf{w}_t^k$	-
Scheme 1	Sahu et al. (2018)	$\mathcal{S}_t \sim \mathcal{W}(N,K,\mathbf{p})$	$\frac{1}{K}\sum_{k\in\mathcal{S}_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[5]
Scheme 2	Ours	$\mathcal{S}_t \sim \mathcal{U}(N,K)$	$\sum_{k\in\mathcal{S}_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})$[6]

$$\mathbb{E}\left[F(w_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B + \boxed{C})}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|w_1 - w^*\|^2\right),$$

***Scheme T* – Balanced data assumption 완화**

$$\tilde{F}_k(w) := \underline{p_k N F_k(w)} \qquad F(w) = \sum_{k=1}^{N} p_k F_k(w) = \boxed{\frac{1}{N}}\sum_{k=1}^{N} \tilde{F}_k(w)$$

Scaling

$$\tilde{L} := \nu L$$

$$\tilde{\mu} := \varsigma \mu$$

$$\nu := N \cdot \max_k p_k \qquad \varsigma := N \cdot \min_k p_k$$

$$\tilde{\sigma}_k := \sqrt{\nu}\sigma_k$$

$$\tilde{G} := \sqrt{\varsigma} G$$

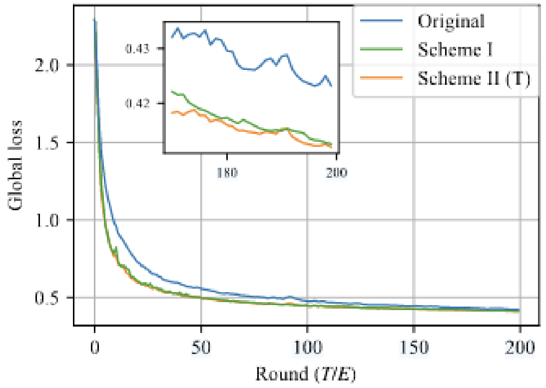
Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Experiments (1)

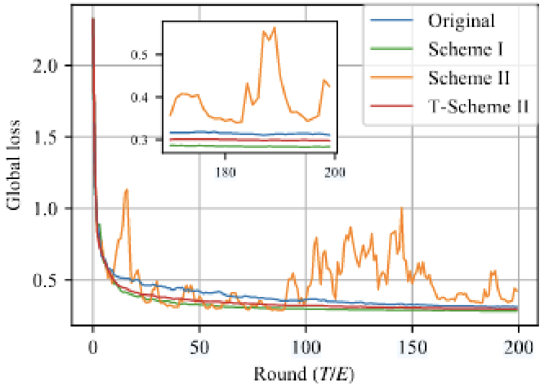
- ➢ 데이터 균형: ***Scheme*** 관계 없이 안정적으로 수렴
- ➢ 데이터 불균형: ***Scheme 1***과 ***Scheme T***가 안정적으로 수렴

데이터 균형



(c) Different schemes

데이터 불균형



(d) Different schemes

Convergence of FedAvg

On the Convergence of FedAvg on Non-IID Data

❖ Hyperparameter E의 영향

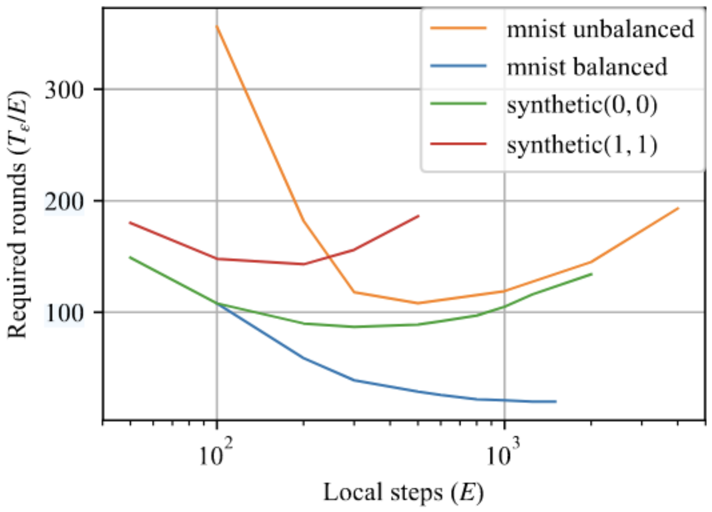
- ➢ T_ϵ: ϵ만큼의 정확도를 달성하는데 필요한 local update 수
- ➢ $\frac{T_\epsilon}{E}$: ϵ만큼의 정확도를 달성하는데 필요한 **communication round** 수

$$\frac{T_\epsilon}{E} \propto \left(1 + \frac{1}{K}\right) EG^2 + \frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + L\Gamma + \kappa G^2}{E} + G^2$$

Communication round를 결정하는 것은 E

E ↓ : *Local model underfitting*

E ↑ : *Local model overfitting*



(a) The impact of E