Gossip Distillation: 学習用データを送信しない 非集中分散学習

森脇　泰介†　　首藤　一幸††

† 東京工業大学　〒 152–8550 東京都目黒区大岡山２丁目１２番１号
†† 京都大学　〒 606–8315 京都府京都市左京区吉田本町
E-mail: †moriwaki.t.aa@m.titech.ac.jp, ††shudo@media.kyoto-u.ac.jp

あらまし　深層学習にて精度の高いモデルを得るには多くの学習データが要る一方で，プライバシなどの理由からデータを 1ヶ所に集められない場合がある．さらに，集中的なサーバを排し全ノードが非集中的に通信することで，耐故障性の向上や，サーバが最初に学習済みモデルを手に入れるという不公平の解消を図れる．従来の分散深層学習では，ノード間でモデルの勾配情報を通信していたが，我々はここに Knowledge Distillation を応用し，ノード群は共通データに対する推論結果を通信する．これにより，通信量が大幅に減り，また，ノードごとに異なるニューラルネットワークを用いることが可能になる．実験においては複数のデータセットに複数の Non-IID パターンを設定し手法を適用したところ，中央集権型の先行研究と同等あるいは上回る精度を得た．

キーワード　非集中分散学習，深層学習，Gossip，Knowledge Distillation

1 はじめに

深層ニューラルネットワークを学習する際には一般的に大規模かつ多様なデータセットが必要になる．これによって，他人に共有することが困難なデータを用いるような学習タスクに取り組むことが困難となる．

そこで，データを直接動かさない Federated Learning [1] が提唱され，現在では医療 [2] や IoT [3] に用いられている．

Federated Learning はデータを動かないという意味では非常に大きな意味を持つが，ユーザーが一方的に巨大なサーバーを所有する企業や団体を信頼しなければならないという制約を持つ．そこで Federated Learning を非集中環境で実行できるようブロックチェーンの考え方を取り込んだ FL-Block [4] や Gossip Learning の学習法を活用した FedGosp [5] が提唱されるなど，非集中分散学習は現代社会において非常に重要になっている．

非集中分散学習を実際のデバイスを使用して行うことを想定すると，ノードデバイスのスペックに応じて使用するモデルを組み合わせることが重要になってくる．そこで本論文では様々なスペックのノードを使用して学習できるような非集中型の分散深層学習を行うことを目標としている．

2 関連研究

2.1 Federated Learning

Federated Learing では中央の巨大なサーバー及びそれと接続された多数のデバイスを用いる．デバイスは携帯端末や IoT 機器など幅広いものを使用できる．Federated Learing では中央の巨大なサーバー及びそれと接続された多数のデバイスを用いる．

一般的な集中学習では，データを中央サーバーに送信して学習を行うが，代わりにモデルの勾配情報を送信し，ノード側のデバイスで学習した後，モデルの勾配を再びサーバーに送付しサーバー側で勾配を統合したものを新しいモデルとする，といったことを繰り返すやり方をとる．

これにより，データ自体を他者に公開する必要がなく，プライバシーを保つことが可能となる．

2.2 Knowledge Distillation

軽量で単純なモデルを用いても高い精度を出すための手法として Knowledge Distillation [6] が挙げられる．これは，大きく構造が複雑であるようなモデルで学習した結果を軽量モデルの学習に利用する，といったものである．

これを用いることによって 0-1 で与えられる通常の学習データよりもより細かなデータを学習に用いることができるため，精度が向上している．

2.3 Decentralized Learning via Adaptive Distillation (DLAD)

DLAD [7] は Federated Learning に Knowledge Distillation の考え方を取り込んだ手法である．DLAD ではノードが所有するプライベートデータセットとは別に共通の蒸留データセットを用いる．Federated Learning においてはサーバーに対して学習したモデルの勾配を送信するが，DLAD ではプライベートデータセットを学習したモデルに追加で蒸留データセットを推論させる．推論結果を中央サーバーに送付し，プライベートデータセットの内容に応じて適応的に統合し，サーバーではその統合された結果が出力できるようなモデルを新たに学習する，といったステップをとる．

このステップをとることによって通信回数は 1 回だけで済み，悪い通信環境下においても正常に作用する他，全てのクライアントのデータを平均的にとるよりも高い精度を出すことが可能

となる．

2.4 gossip を活用した分散機械学習

gossip はネットワーク上で情報をマルチキャストする方法の一つで P2P ネットワークや分散ネットワークでよく使用される手法 [8] である．gossip を用いた分散学習である Gossip Learning [9] も存在する．これはプライバシーを考慮し，データをノードから移動させず学習を行う手法であり，通信方法に gossip を用いている．また，Gossip Learning を改良し，ノードの次数に応じて選択確率を操作した手法 [10] や通信頻度を変更することにより輻輳を解消した手法 [11] も提案されている．

gossip を Federated Learning に取り組んだ手法 [12] もある．具体的には，複数のクライアントが所持するモデルの重みをを gossip の手法を用いて合成し、組み合わせている．したがって，gossip は非集中分散学習を行う上で非常に有用な手法となりうると考えられる．

gossip を活用した FedGosp [5] という手法によってクライアント間での通信も行う試みがなされている．FedGosp では，中央サーバーによってクラスタリングされた，それぞれ類似した重みを所有するクラスタ間で通信をし、更新されたパラメータを中央サーバーへ送信し、統合を行うといった形をとっており、Federated Learning のクライアント学習部分に非集中学習を織り込んだ形になっている．その中の通信においては gossip を用いている。

3 Gossip Distillation

この章では，提案手法である Gossip Distillation について説明する．

3.1 問 題 設 定

$x \in \mathcal{X}$ を入力サンプル (画像など)，$y \in \mathcal{Y}$ を入力サンプルにおけるラベルとする．この研究においては特に $\mathcal{Y} = \{1, 2, \ldots, N\}$ の N クラス分類問題について考えていく．

分類学習における問題を定式化するために，複数のクライアントが存在することを考える．

N 個のノード $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_N$ がそれぞれラベル付きのデータセット $\mathcal{D}_i = (\mathcal{X}_i, \mathcal{Y}_i)\ i \in \{1, 2, \ldots, N\}$ を所持しているとする．ここで，$\mathcal{X}_i = \{x_j^{(i)}\}$，$\mathcal{Y}_i = \{y_j^{(i)}\}$ であり，$y_j^{(i)}$ は $x_j^{(i)}$ にアノテーションされたラベルである．

$\mathcal{D}_i$ は $\mathcal{N}_i$ のみが見ることができ，他のノード $\{\mathcal{N}_j | j \neq i\}$ から見ることはできない．

以下，D_i のことを Private Data (PD) と呼ぶ．

また，各ノードは PD を用いて学習を行った独自のモデル $\mathcal{M}_i : \mathcal{X} \to \mathcal{Y}_i$ を所持している．

現実のデータは Non-IID であることが十分に想定できるため，PD も Non-IID である．そのため，$\mathcal{M}_i$ と $\mathcal{M}_j\ (i \neq j)$ を用いて同一の画像を推論しても必ずしも同じ結果になるとは限らない．

我々の目標は，相互の分類能力を互いに活用することによって $\mathcal{M}_i$ 及び $\mathcal{M}_j$ の出力結果を同一に，そして正しく推論することである．

3.2 蒸　　留

ノードが所有する $\mathcal{M}_i$ から他のモデルに分類能力を移行させるために，Knowledge Distillation [6] の考え方を利用する．

本研究においては PD とは別にどのノードも見ることができるラベルなしのデータセット，Distillation Dataset (DD) が存在する．DD の作成のため、PD とは別のデータ群である，Common Data $\mathcal{X}_c$ (CD) を準備する．

DD を用いることによって，PD で学習させた，ノードが所有するモデルにしかない分類能力を他のモデルにも伝搬させていく．ただ，多数のノードの知識を同時に送受信することは困難となるため，gossip (pull) の手法を用いて 1 対 1 で通信するということを繰り返していく．

具体的には，あるノード $\mathcal{N}_{tgt}$ が所有するモデル $\mathcal{M}_{tgt}$ が $\mathcal{X}_c$ を推論したリスト $\mathcal{L}_{tgt}$ 及び，現在までにデータを送信した回数 C_{tgt} を受信したノード $\mathcal{N}_{get}$ は自信が所有する $\mathcal{X}_c$ を推論したリスト $\mathcal{L}_{get}$ と現在までにデータを送信した回数 C_{get} を受信した $\mathcal{L}_{tgt}$ と C_{tgt} と加重平均

$$\mathcal{L}_{mix}^{(i)} = \frac{\mathcal{L}_{tgt}^{(i)} \times C_{tgt} + \mathcal{L}_{get}^{(i)} \times C_{get}}{C_{tgt} + C_{get}}\ i \in \{1, 2, \ldots, n\} \tag{1}$$

で統合をする．ここで n は DI の総数である．その後，式 1 によって生成された $\mathcal{L}_{mix}$ の要素のリストの最大値を要素とするラベルリスト $\mathcal{Y}_c$ を作成する．すなわち $\mathcal{Y}_c^{(i)} = \max \mathcal{L}_{mix}^{(i)}$ となる．その後画像を $\mathcal{X}_c$，ラベルを $\mathcal{Y}_c$ とする DD $\mathcal{D}_{dist} = (\mathcal{X}_c, \mathcal{Y}_c)$ を作成する．

DD は通信を重ね他の推論リストを取り込むたびに再生成を重ね，また再生成を重ねた DD を用いて再びモデル $\mathcal{M}_i$ を学習していくため，最終的には PD 全てのデータの分類能力を各ノード画所有する $\mathcal{M}_i$ が獲得することが期待される．

3.3 具体的な手法

3.3.1 ローカルモデルの学習

はじめに、モデル $\mathcal{M}_i$ に各ノードが所有する PD_i を学習させた．

その後 $\mathcal{M}_i$ に Distillation Dataset を推論させた後，推論結果を各クラスごとに保持する．

具体的には、アルゴリズム 1 の通り．

Algorithm 1 Gossip Distillation (local phase)

Input: Each model M_i, Each Private Data PD_i, Common Data $\mathcal{X}_c$, number of Nodes N

Output: Each Inference List IL_i, Trained Weight W_i

```
function LOCAL ROUND(W_0, PD_i, DRS)
    for all each client number do
        W_i ← Update(W_0, PD_i)
        IL_i ← Inference(W_i, X_c)
    end for
end function
```

3.3.2 グローバルフェーズ

3.3.2 節では，モデル M_i を 3.3.1 節で生成した IL_i をノード間

でやりとりし，DD を作り，学習させていくことによってアップデートしていく．その際，設定したラウンド数以下のことを繰り返していく．

（1） 自分以外の任意のノードを指定

（2） そのクライアントの推論リスト，及び指定したクライアントの本ループ実行回数を受信する

（3） 式 1 に則り，DD を作成する

（4） DD を用いて自分が所有するモデルを学習

（5） 学習したモデルを用いて DD の画像を推論することによって送信用の推論リストを更新する

具体的にはアルゴリズム 2 の通り．

Algorithm 2 Gossip Distillation (gossip phase)

Input: Each Trained Weight W_i, Round count R, Target Node N_{tgt}, User Node N_{usr}

Each Node has set $N = (\mathcal{L}, C)$

function GlobalRound(R, N_{usr})

for all Rounds $r = 1, 2, \ldots, R$ **do**

$tgt \leftarrow$ Randomly selected node id

download $N_{tgt} = (\mathcal{L}_{tgt}, C_{tgt})$

function make Distillation Dataset(N_{tgt}, N_{usr})

image…Images for Distillation Dataset

label…make as follows

1. Integrate like Equation 1 by using (N_{tgt}, N_{usr})

2. The highest-valued index of each element is the label.

return $DD_i = (image, label)$

end function

$W_i \leftarrow Update(W_i, DD_i)$

$IL_i \leftarrow Inference(W_i, DRS)$

end for

end function

4 実　　験

本章では，提案手法である Gossip Distillation を用いた実験を行っていく．その際に集中分散学習で Knowledge Distillation を用いている DLAD [7] を先行研究とし，比較を行っていく．

4.1 データセット

本節では使用するデータセットについて説明する．本実験では DLAD に合わせて MNIST, CIFAR-10, CINIC-10 [13] を使用する．CINIC-10 は CIFAR-10 と ImageNet から抽出された計 27 万という巨大なサイズのデータセットであるため，難易度が高いデータセットである．

いずれのデータセットにおいても，Training Dataset の内 20% を Private Data として，残りの 80%を Distillation Dataset として用いる．これはラベル付きデータよりもラベルなしデータの方が多い実世界の状況を模擬している．

PD で用いるデータセットに関しては，ランダムにピックアップしているが実験段階では乱数を固定し同分布になるようにしている．また，PD に関しては n 個のクライアントで分割して使用している．これによって各ノード N_i はデータセット $\mathcal{D}_i = (\mathcal{X}_j, \mathcal{Y}_j)_{j=1}^{n/i}$ を有する．

本実験においては，データの重複がないように MNIST なら 60000，CIFAR-10 ならば 50000 のうちの 20%のデータを分割して使用している．DLAD では重複を許容しているが，本実験では重複を許容していない．そのため，各クライアントが所有している PD のサイズは訓練用データセットの約 2%程度になっている．

CINIC-10 に関しては訓練用データセットとバリデーションデータセットがそれぞれ 90000 ずつ用意されている．なのでバリデーションデータセットを Distillation 用のものとして使用しても良いが，本研究においては MNIST，CIFAR と同様の比率である方が前述の状況を作成しやすいと考え，訓練用データセット 90000 を分割して使用している．

表 1 に示すように，5 種類の異なるデータ分散でテストを行った．これに関しては DLAD と同様の分布の他，更にデータの分布がクライアントごとにバラバラになるような 1 分布を追加して行っている．

IID　全てのクライアントが確率 $p_i = [0.1, 0.1, 0.1, \ldots, 0.1]$ でデータを所有している．

Non-IID #1　各クライアントが 10 クラスのうちの 2 クラスのデータを所有している．すなわち，$p_{5k+1} = [0.5, 0.5, 0, 0, \ldots, 0], p_{5k+2} = [0, 0, 0.5, 0.5, 0 \ldots, 0] \ldots$ となるようにしている．

Non-IID #2　クラス 0-4 に関しては全てのクライアントが有しているが，5-9 に関してはいずれの 1 つしか所有していない．すなわち，$p_{5k+1} = [0.1, 0.1, \ldots, 0.1, 0.5, 0, \ldots, 0], p_{5k+2} = [0.1, 0.1, \ldots, 0.1, 0, 0.5, 0, \ldots, 0], \ldots$

$p_{5k+5} = [0.1, 0.1, \ldots, 0.1, 0, \ldots, 0, 0.5]$ となるようにしている

Non-IID #3　各クライアントがそれぞれ 4 つのクラスを有しており，各クラスはそれぞれ異なる 2 クライアントグループが有している．すなわち

$$p_k = \begin{cases} 0.25 & \text{(class include in private client)} \\ 0 & \text{(otherwise)} \end{cases}$$

どのクラスをノードクライアントが所有しているかは表 1 を参照のこと．

Non-IID #4　各クライアントがそれぞれ 1 つのクライアントのみ有しており，先行研究よりもよりストイックな状況を想定している．すなわち $p_{10k+1} = [1, 0, \ldots, 0], p_{10k+2} = [0, 1, 0, \ldots, 0], \ldots, p_{10k+10} = [0, \ldots, 0, 1]$ となるようにしている．

表 1　実験に用いたデータ分散

	U_{5n+1}	U_{5n+2}	U_{5n+3}	U_{5n+4}	U_{5n+5}
IID	0 – 9	0 – 9	0 – 9	0 – 9	0 – 9
Non-IID #1	0, 1	2, 3	4, 5	6, 7	8, 9
Non-IID #2	0 – 4, 5	0 – 4, 6	0 – 4, 7	0 – 4, 8	0 – 4, 9
Non-IID #3	0, 1, 2, 3	0, 4, 5, 6	1, 4, 7, 8	2, 5, 7, 9	3, 6, 8, 9
Non-IID #4			i (U_i)		

以上から、PD のみを用いて学習を行った各ノードが所持するモ

デルの精度の理論値は (IID, Non-IID #1, Non-IID #2, Non-IID #3, Non-IID #4) = $(1, 0.2, 0.6, 0.4, 0.1)$ となるはずである。

4.2 モ デ ル

本実験で使用するクライアントモデルとしてメインに Deep Residual Network (ResNet) [14] を使用する．また，異なるモデルを使用実験においては Densely-connected Convolutional Networks (DenseNet) [15] 及び MobileNet V3 (small) [16] を利用している．

前者については DLAD との比較のために用いているが，後者においては，ResNet-18 と比較してネットワークが軽量かつ単純なモデルを併用することでの影響について調べるためである．

本実験で使用する 3 モデルのサイズにおいては表 2 の通り．

表 2 使用モデル (サイズはモデル内パラメータの容量)

model	params	size(MiB)
ResNet-18	11,689,512	49.03
DenseNet-121	7,978,856	33.47
MobileNet v3(small)	2,542,856	10.66

各モデルにおいては，予め事前学習した重みを採用し，その後に SGD Optimizer を用いて学習率 5×10^{-6} で 200 エポック，またサイズ 40 のミニバッチで local round の学習を行った。また，グローバルラウンドでの各ステップにおける学習は 1 エポックのみ行っていく．学習率及びミニバッチサイズはローカルモデルの学習時と同じ条件に設定をした．

4.2.1 実 験 環 境

本実験で使用するマシンの具体的なスペックは表 3 の通り．今回は全ての環境をこのマシンで行う，つまり全てシミュレーション下で行っている．

表 3 本実験で使用するマシンの詳細

	spec
OS	Ubuntu 20.04.2 LTS
CPU	Intel Xeon Platinum 8368 Processor (38core, 2.4GHz) × 2
GPU	NVIDIA Tesla A100-PCIE-40GB

4.3 実 験 内 容

本実験ではまず，DLAD と同様の条件で精度を観測し，その差異を検証する．次に，上記で説明した通り，ResNetr-18 と DenseNet-121, MobileNet v3 (small) をそれぞれ組み合わせた実験を行っていき、精度の比較及び検証を行っていく．また、その際にグローバルラウンドで送信される IF_i のサイズを測定する．これによってモデルの重みを送信する Federated Learning 等手法と送信するサイズの比較も行っていく．

4.4 結　　果

表 4 及び図 2 は各データセット及び手法によって ResNet-18 を学習したときの分類精度を示している．値は，10 ノードにおける精度の中央値を表している．また、DLAD と Gossip Distillation において高い値が出ている方を太字にしている．

まず，全体として local phase での各モデルの精度よりも global phase での精度の方が全ての実験において高いことから，蒸留をすることの意味があることが示されている．また，global phase における精度を DLAD やデータが全て一箇所にある場合 (labeled) と比較したところ，DLAD とほぼ同等，時には上回る結果を得ることができている他，labeled とも十分対抗でき得ることが分かった．

この結果によって，中央集約する既存の手法に代替し得るだけの性能を持つことが確認された．

local phase においては PD が所有するデータ数の少なさからか精度が低いが，各モデルの PD に対する精度は IID や Non-IID に関係なく高かった (MNIST においては 99%以上，CINIC-10 においても 75%以上)．そのためか，global phase において情報伝達が始まってすぐのラウンドでテストデータに対する精度の著しい向上が見えた．

また，同じデータセット内の Non-IID4 パターンにおける精度を比較してもほぼ同程度の結果が出ていることが分かる．このことから，ラウンドを重ねるごとに各ユーザーが所有している PD に関係なく全ての知識が行き渡っていると推測できる．図 1 は表 4 内の CIFAR-10 の IID, Non-IID #1~3 について，local phase 終了時の精度及び global phase について精度をプロットしたグラフであるが，全てにおいて早いラウンド時点で精度が上昇しており Gossip Distillation の有意性が示されている．

表 5 においては，複数のモデルを用いた結果を示している．こちらに関しては，ResNet-18 と他のニューラルネットワークをそれぞれ 5 ノードずつ用いて学習を行った．表 5 はその 5 ノードずつの精度の中央値を表している．DenseNet-121 を組み合わせた結果を見ていくと，DLAD の CIFAR-10 においての結果と比較しても高い結果が出ていることが分かる．他のデータセットを用いた例においても，ResNet-18 のみを用いた場合と同等，もしくは若干だが高い精度が出ている．

また，mobilenet V3(small) を組み合わせた結果を見ていくと，こちらも ResNet-18 のみを用いた例と同程度の精度の高さが出ている．また，mobilenet を組み合わせた結果を見ると，ResNet よりかは精度が落ちてしまうもののこちらも Gossip Distillation による精度が出ていると言える．モデルを mobilenet10 個に置き換えた以外は同条件で実験を行ったところ，精度が 0.47 程度であったため，複数のモデルを組み合わせても同程度の精度が出ると言える．このことから Gossip Distillation は別々のモデルを組み合わせて行うことが十分に可能であると判断した．

また、表 6 は各データセット使用時の送信用 Common Data のサイズを表している．例えば ResNet-18 のモデルの勾配パラメータのサイズは 2 によると 49.03MiB であることから、全てのデータセットにおいて 10%前後に削減できていると言える．

このことから、Gossip Distillation は従来のモデルの勾配を送るような非集中分散学習よりもより軽量なデータを送受信するため、通信環境がより悪い状態でも実行できるといえる．

表 4　データセット及び分布に対する精度比較 (モデルは全て ResNet-18, $n = 10$)

Dataset	MNIST					CIFAR-10					CINIC-10				
Distribution	IID	NIID1	NIID2	NIID3	NIID4	IID	NIID1	NIID2	NIID3	NIID4	IID	NIID1	NIID2	NIID3	NIID4
Local round	0.7089	0.1861	0.4698	0.3504	-	0.3870	0.1669	0.2693	0.2798	-	0.3596	0.1592	0.2536	0.2609	-
DLAD [7]	**0.9821**	**0.9820**	**0.9828**	**0.9840**	-	**0.7314**	0.6657	0.6847	0.7027	-	**0.6323**	**0.6266**	0.5666	0.5934	-
labeled [7]	0.9836	0.9868	0.9845	0.9857	-	0.7115	0.8127	0.7576	0.7755	-	0.6256	0.6880	0.6183	0.6574	-
Gossip Distillation	0.9644	0.9647	0.9669	0.9680	0.9715	0.7187	**0.7159**	**0.7181**	**0.7226**	0.7280	0.5935	0.6096	**0.5973**	**0.6064**	0.6157

表 5　複数モデルを組み合わせた時の精度の比較 ($n = 10$)

Main Model		ResNet-18					
Sub Model		DenseNet-121			MobileNet v3(small)		
Dataset		MNIST	CIFAR-10	CINIC-10	MNIST	CIFAR-10	CINIC-10
Non-IID # 1	DLAD [7]	– / –	0.6657 / 0.6642	– / –	– / –	– / –	– / –
Res/other	Gossip Distillation	0.9709 / 0.9781	**0.7279** / **0.7403**	0.6096 / 0.6180	0.9718 / 0.9540	0.7265 / 0.5550	0.6077 / 0.4796

表 6　データセットごとの DD のサイズ

	dataset	size (MiB)
toprule	MNIST	3.46
	CIFAR-10	2.88
	CINIC-10	5.18

5 ま と め

本研究では，非集中分散学習を行い，更には使用デバイスの性能に応じて様々なモデルを使用できるような手法を提案した．実験の結果，既存の中央集権型の分散学習と同等の精度を出すことができた他，モデルが単純で浅いものが含まれていてもそうでない条件と同等の精度まで向上できるということが分かった．これによって，中央の高性能のサーバーがなくともデータを動かさずに学習を行い，かつモデルの勾配よりも容量が軽い Distillation Dataset の推論結果を送受信するためより学習までの準備・コストが軽量になるであろう．

本研究においては少数のデバイスを使用することをシミュレートしたが，使用デバイスの総数が増加しても十分に行うことは可能だと考えている．今後はデバイスネットワークをより巨大に，またデバイス間の信用が困難である不完全ネットワークを想定した実験を行うことによってより実社会に近い環境をシミュレートしていく．

謝　　辞

本研究を遂行するにあたり，オムロンサイニックエックス株式会社 米谷竜氏においては，提案内容に関して議論していただき感謝の意を表す．また，データ活用社会創成プラットフォーム mdx [17] を使用して本研究を実施した．

References

[1] H. B. McMahan, E. Moore, D. Ramage, *et al.*, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *Proc. AISTATS 2017*, vol. 54, 20–22 Apr 2017, pp. 1273–1282.

[2] A. Rauniyar, D. H. Hagos, D. Jha, *et al.*, “Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions,” 2022.

[3] D. C. Nguyen, M. Ding, P. N. Pathirana, *et al.*, “Federated learning for internet of things: A comprehensive survey,” *Proc. IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021.

[4] Y. Qu, L. Gao, T. H. Luan, *et al.*, “Decentralized privacy using blockchain-enabled federated learning in fog computing,” *Proc. IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5171–5183, 2020.

[5] G. Li, Y. Hu, M. Zhang, L. Li, T. Chang, and Q. Yin, “FedGosp: A Novel Framework of Gossip Federated Learning for Data Heterogeneity,” in *Proc. IEEE SMC 2022*, 2022, pp. 840–845.

[6] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[7] J. Ma, R. Yonetani, and Z. Iqbal, “Adaptive distillation for decentralized learning from heterogeneous clients,” in *Proc. ICPR 2020*, IEEE, 2021, pp. 7486–7492.

[8] S. Maarten van and T. Andrew S., *Distributed Systems Third edition*. 2020.

[9] R. Ormándi, I. Hegedűs, and M. Jelasity, “Gossip learning with linear models on fully distributed data,” *Concurrency and Computation: Practice and Experience*, vol. 25, no. 4, pp. 556–571, 2013.

[10] Y. Takahashi and K. Shudo, “P2P ネットワーク上のデータに対する偏りのない機械学習手法,” *DEIM Forum 2017*, Feb. 2017.

[11] H. Oguni and K. Shudo, “Addressing the heterogeneity of A Wide Area Network for DNNs,” in *Proc. IEEE CCNC 2021*, 2021, pp. 1–6.

[12] C. Hu, J. Jiang, and Z. Wang, “Decentralized Federated Learning: A Segmented Gossip Approach,” *arXiv e-prints*, arXiv:1908.07782, Aug. 2019.

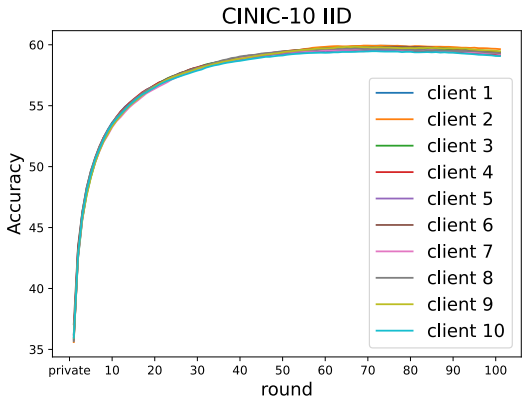
[13] L. N. Darlow, E. J. Crowley, A. Antoniou, *et al.*, “CINIC-10 is not ImageNet or CIFAR-10,” *arXiv e-prints*, arXiv:1810.03505, Oct. 2018.

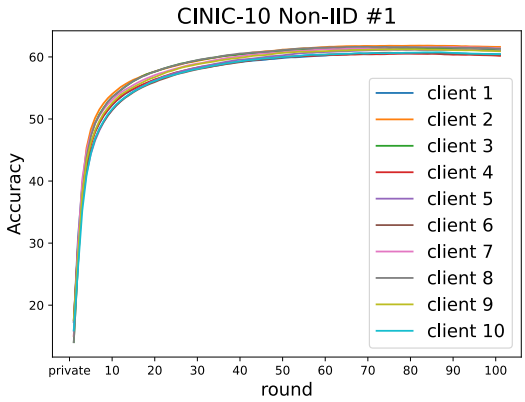
[14] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE CVPR 2016*, 2016, pp. 770–778.

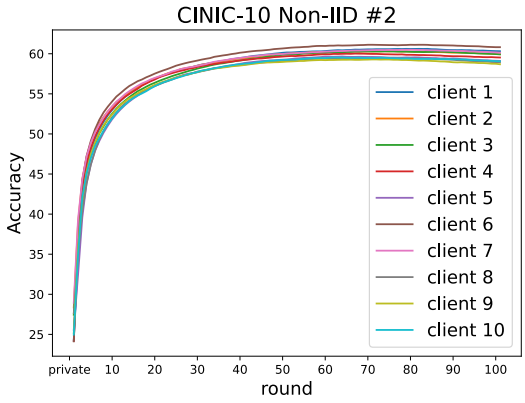
[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc. IEEE CVPR 2017*, 2017, pp. 4700–4708.

[16] A. Howard, M. Sandler, G. Chu, *et al.*, “Searching for mobilenetv3,” in *Proc. IEEE CVPR 2019*, 2019, pp. 1314–1324.

[17] T. Suzumura, A. Sugiki, H. Takizawa, *et al.*, “mdx: A Cloud Platform for Supporting Data Science and Cross-Disciplinary Research Collaborations,” in *Proc. IEEE DASC 2022*, 2022, pp. 1–7.







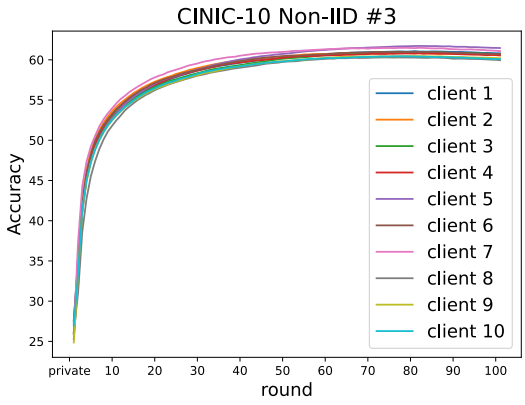


図 1　CINIC-10 及び ResNet-18 を用いて Gossip Distillation を行った時の IID, Non-IID3 つの Accuracy の private phase 及び gossip phase の round ごとの推移を表したグラフ

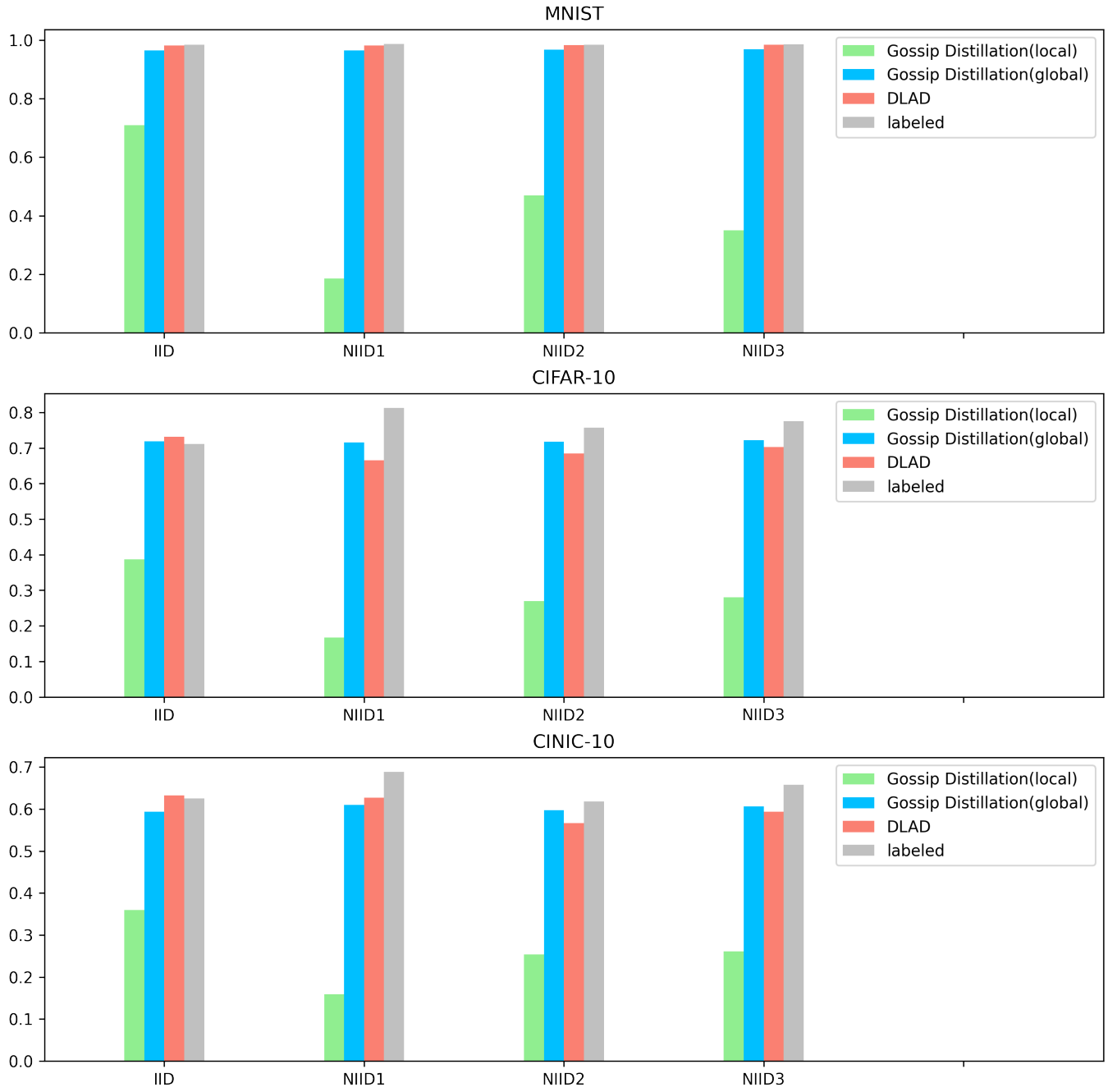


図 2　データセット及び分布に対する精度比較グラフ．local, global はそれぞれのフェーズ終了時の各ノードの Accuracy の中央値である