Université de Montréal

Generalization in Federated Learning

par

Irene Tenison

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en Discipline

August 30, 2022

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

Generalization in Federated Learning

présenté par

Irene Tenison

a été évalué par un jury composé des personnes suivantes :

Guillaume Rabusseau
(président-rapporteur)

Irina Rish et Eugene Belilovsky
(directeur de recherche)

Laurent Charlin
(membre du jury)

Résumé

L'apprentissage fédéré est un paradigme émergent qui permet à un grand nombre de clients disposant de données hétérogènes de coordonner l'apprentissage d'un modèle global unifié sans avoir besoin de partager les données entre eux ou avec un stockage central. Il améliore la confidentialité des données, car celles-ci sont décentralisées et ne quittent pas les dispositifs clients. Les algorithmes standard d'apprentissage fédéré impliquent le calcul de la moyenne des paramètres du modèle ou des mises à jour du gradient pour approcher le modèle global au niveau du serveur. Cependant, dans des environnements hétérogènes, le calcul de la moyenne peut entraîner une perte d'information et conduire à une mauvaise généralisation en raison du biais induit par les gradients dominants des clients. Nous supposons que pour mieux généraliser sur des ensembles de données non-i.i.d., les algorithmes devraient se concentrer sur l'apprentissage du mécanisme invariant qui est constant tout en ignorant les mécanismes parasites qui diffèrent entre les clients.

Inspirés par des travaux récents dans la littérature sur la distribution des données, nous proposons une approche de calcul de la moyenne masquée par le gradient pour FL comme alternative au calcul de la moyenne standard des mises à jour des clients. mises à jour des clients. Cette technique d'agrégation des mises à jour des clients peut être adaptée en tant que remplacement dans la plupart des algorithmes fédérés existants. Nous réalisons des expériences approfondies avec l'approche de masquage du gradient sur plusieurs algorithmes FL avec distribution, monde réel et hors distribution (en tant qu'algorithme fédéré). hors distribution (comme le pire des scénarios) avec des déséquilibres quantitatifs. déséquilibres quantitatifs et montrent qu'elle apporte des améliorations constantes, en particulier dans le cas de clients hétérogènes. clients hétérogènes. Des garanties théoriques viennent étayer l'algorithme proposé.

Mots clés : Apprentissage fédéré, généralisation hors distribution

Abstract

Federated learning is an emerging paradigm that permits a large number of clients with heterogeneous data to coordinate learning of a unified global model without the need to share data amongst each other or to a central storage. In enhances data privacy as data is decentralized and do not leave the client devices. Standard federated learning algorithms involve averaging of model parameters or gradient updates to approximate the global model at the server. However, in heterogeneous settings averaging can result in information loss and lead to poor generalization due to the bias induced by dominant client gradients. We hypothesize that to generalize better across non-i.i.d datasets, the algorithms should focus on learning the invariant mechanism that is constant while ignoring spurious mechanisms that differ across clients.

Inspired from recent works in the Out-of-Distribution literature, we propose a gradient masked averaging approach for FL as an alternative to the standard averaging of client updates. This client update aggregation technique can be adapted as a drop-in replacement in most existing federated algorithms. We perform extensive experiments with gradient masked approach on multiple FL algorithms with in-distribution, real-world, and out-of-distribution (as the worst case scenario) test datasets along with quantity imbalances and show that it provides consistent improvements, particularly in the case of heterogeneous clients. Theoretical guarantees further supports the proposed algorithm.

Keywords: Federated Learning, Out-of-Distribution Generalization

Contents

Résumé 5

Abstract 7

List of tables 11

List of figures 13

List of acronyms and abbreviations 15

Acknowledgements 17

Introduction 19

Contribution 20

Outline 20

Working Paper 21

Funding Acknowledgment 21

Chapter 1. Background 23

1.1. Introduction to Federated Learning 23

1.2. Federated Learning Settings 25

1.3. Related Works 26

Chapter 2. OOD Generalization and Federated Learning 29

2.1. Out-of-Distribution Generalization 29

2.2. AND-Mask for OOD Generalization 31

2.3. Connections between OOD Generalization and FL 32

Chapter 3. Gradient Masked Averaging 35

3.1. Gradient Masked Averaging 35

3.1.1. Motivation 35
3.1.2. Binary AND-Mask to Soft Mask 36
3.1.3. Algorithm 37
3.1.4. Effect of τ 40
3.2. Federated Aggregation Preliminaries 41
3.3. Theoretical Guarantees 41
Chapter 4. Experiments 47
4.1. Real-World Evaluation 48
4.2. Out-of-Distribution Evaluation 49
4.3. In-Distribution Evaluation 50
4.4. Quantity skew 52
4.5. Convex Objective 54
4.6. Client Momentum and Group Norm 54
4.6.1. Effect of Client Momentum 55
4.6.2. Effect of GroupNorm 56
4.7. Membership Inference Attack 56
4.8. Conclusion and Future Work 58
References 59
Appendix A. Learning Rates 63

List of tables

4.1 Real-World evaluation on FEMNIST and Out-of-Distribution evaluations with and without label distribution skew on FedCMNIST and FedRotMNIST. Average best test performance(%) over 4 independent runs of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions are reported below. The best average result among AVG and GMA having atleast 0.01% higher than the other algorithm is shown in bold. 48

4.2 This table shows the label distribution across clients for an IID setting. Each client will have randomly chosen examples from all 10 classes. This represent the 10 class setting in MNIST. We have considered 3 clients for the table. The same pattern would be present across all clients. 51

4.3 This table shows the label distribution skew used by us for our experiments for a non-IID data distribution across clients. This represents the 10 class setting in MNIST. We have taken 3 clients. The same pattern would be present across all clients. 51

4.4 In-Distribution evaluations with and without label distribution skew on MNIST, FMNIST, FEMNIST, and CIFAR10 and Quantity skew based evaluation on CIFAR10. Average best test performance(%) over 4 independent runs of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions are reported below. The best average result among AVG and GMA having atleast 0.01% higher than the other algorithm is shown in bold. 52

4.5 Performance of FedAVG, FedProx, FedAdam, and FedYogi on CIFAR10 and CIFAR100 when C clients are sampled from N for training in each communication round. 53

4.6 Performance of FedAVG across a range of global learning rate and client rate on non-iid FMNIST. It can be observed that GMA outperforms AVG in most of the cases where the algorithms learn and converge. 53

4.7 Performance of the algorithms and their GMA versions with and without momentum(ρ) on non-i.i.d distributed FMNIST using an LeNet model. Momentum

improves performance of the algorithms. Irrespective of momentum, GMA outperforms AVG. 55

4.8 Average in-distribution test performance(%) over the last 10 communication rounds of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions on i.i.d and non-i.i.d distributions of CIFAR-10 on ResNet18 models using batch normalization and group normalization. The best result among AVG and GMA versions of each algorithm is shown in bold. 56

A.1 The best learning rates corresponding to the performances of the algorithms and datasets as reported in Table 4.1 63

A.2 The best learning rates corresponding to the performances of the algorithms and datasets as reported in Table 4.4 64

List of figures

1.1 Cross-Device Federated Learning[**25**] 24

3.1 Fraction of clients having *agreement* $< \tau(= 0.4)$ vs. homogeneity of data 36
3.2 FedAVG[**36**] vs. FedGMA vs. Binary Mask [**40**] on MNIST with label skew. 36
3.3 Effects of τ on model performance 40

4.1 Train accuracy and test accuracy vs. communication rounds of gradient masked and naive averaging versions of the algorithms on FedCMNIST distributed non-i.i.d across clients. 49
4.2 (a) Test accuracy vs. Number of selected clients in the federated network. (b) Test accuracy vs. number of local epochs per client in each communication round. The experiment was on non-i.i.d distributed FMNIST using a LeNet model. 54
4.3 (a) Data split and creation for attacker model (b) Test loss vs. epochs of the logistic regression attacker model. 57

List of acronyms and abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DL	Deep Learning
ML	Machine Learning
DP	Differential Privacy
IoT	Internet of Things
SGD	Stochastic Gradient Descent
FL	Federated Learning
IID	Independent and Identical Distribution
OOD	Out-Of-Distribution

Acknowledgements

Throughout my masters, I was supported and encouraged by numerous people. First, I would like to thank my supervisors. Dr. Irina Rish, for welcoming me to her group, trusting my abilities, and supporting me by all means. Dr. Eugene Belilovsky, for his mentorship, encouragement, and academic advice. Your feedback was very insightful and your comments prompted me to refine my thinking.

I would also like to thank Dr. Edouard Oyallon, who collaborated on the research project. Thank you for supporting the project. I am grateful for your guidance and insights.

Next, I would like to thank collaborators Sai Aravind Sreeramadas and Dr. Vaikkunth Mugunthan, who made this work possible. They devoted time and effort to the research project. It was fun working with them.

None of this would have been possible without the constant support and encouragement of my family and friends. They were tremendously supportive during the most stressful times. Thank you for being there for me throughout this journey.

Introduction

Recently, phones, tablets, smart home assistants, and other personal devices have become the primary source of compute and data collection for most people around the world [**36**]. These devices are equipped with powerful sensors like cameras, microphones, and GPS and are often synchronized with the user's calendars, social-media, and other contacts. These features combined with the fact they are often carried by the user, provides an access to an abundance of data, majority of which is personal and private in nature. With the increasing privacy concerns and constraints, utilizing these data to learn machine learning models in the traditional setting, where the data is collected at a single centralized storage and it is directly accessed by the model, may not be feasible. Federated Learning is a machine learning paradigm that enables learning the decentralized data without having direct access to the raw data; thereby ensuring data privacy.

Federated Learning (FL) is a distributed machine learning approach that allows clients with decentralized data to efficiently learn a shared global model without having to share their sensitive datasets [**36, 25**]. This enhances privacy as data is neither collected at a central location or cloud nor communicated over any channel. Furthermore, [**41**] and [**44**] argue that federated learning has a lower carbon footprint than traditional machine learning. A challenge in FL is heterogeneity in the data distributed across clients. The non-i.i.d data distribution degrades the performance of federated learning models [**34, 59, 70**]. One of the reasons for this is the loss of information regarding invariances across clients induced by the averaging of model parameters or updates. This is further exacerbated by the multiple local steps taken by each client with the aim of reducing communication rounds which results in "client drift"[**26**]. Each client after multiple local steps can progress too far towards minimizing their local objective which may deviate from that of the global objective.

Recently [**40**] proposed an approach for improving generalization across "environments" in the Out-of-Distribution (OOD) setting . In this work, we draw connections between the OOD setting and the federated learning setting, proposing to adapt the approach of [**40**] to FL. Specifically, we propose a new aggregation method called gradient masked

averaging with the goal of improving generalization across clients and of the global model. The gradient masked averaging can be plugged into any FL algorithm as an alternative to naive averaging of model parameters at the server. Intuitively, gradient masking prioritizes gradient components that are aligned with the overall dominant direction across clients while the inconsistent components of the gradient are given lesser importance.

Applying this approach leads to improved performance in the out of distribution FL evaluation settings such as real-world federated EMNIST [**10**] and FedCMNIST and FedRotMNIST [**16**]. We also observe that the robust features of this method leads to improved performance in a variety of other non-iid training scenarios of FL like label skew and quantity skew in a variety of datasets including CIFAR10, CIFAR100, MNIST, and Fashion MNIST.

Contribution

In this thesis, we propose a gradient masked averaging as a drop-in alternative to naive averaging of model parameters or updates to obtain a global model at the server in federated learning. Our main findings and contributions can be summarized as follows:

- We draw connections between OOD generalization in a centralized setting and global model generalization in FL in terms of clients and environments and also in terms of learning objectives.
- We introduce gradient masked averaging as an alternative to naive averaging of parameters or updates in federated algorithms that focus on global model performance.
- We show that the proposed aggregation leads to an algorithm that benefits from standard convergence results in FL.
- We empirically show that applying gradient masking to any FL algorithm consistently improves out-of-distribution generalization performance of the algorithm. This improvement was also observed on in-distribution evaluations and its various settings.

Outline

The thesis is organized as follows. Chapter 1 provides the background information needed to understand the fundamentals of federated learning. We describe the motivations for FL, major problems of the research space, and the various federated settings. We further describe the various federated algorithms. In Chapter 2, we introduce Out-of-Distribution Generalization in centralized learning and draw connections between OOD generalization and

federated learning. In Chapter 3, we introduce the proposed gradient masking algorithm. We also include theoretical guarantees for the same. Chapter 4 is an extensive empirical analysis of the proposed algorithm. We apply the proposed masking upon various FL algorithms like FedAVG, FedProx, SCAFFOLD, FedAdam, and FedYogi with various datasets and data distribution strategies.

Working Paper

This thesis is based on the working paper called Gradient Masked Averaging for Federated Learning[**52**], which is currently under review for NeurIPS 2022. As the first co-author of the paper, I contributed to literature review, implementation, experiments, and paper writing. Dr. Edouard Oyallon contributed to the theoretical guarantees. Further, earlier versions of this work has also previously been presented at ICLR 2021 DPML workshop [**51**] and NeurIPS 2021 PPML workshop [**53**]. The code accompanying the work is available on https://github.com/arvi797/FL. Section **??** is based on the paper Towards Causal Federated Learning For Enhanced Robustness and Privacy[**17**], which was presented at ICLR 2021 DPML workshop. I contributed to the experiments and analysis.

Funding Acknowledgment

During my master's degree, I have received Bourse C scholarship, Student life Improvement scholarship, and Excellence scholarship from University of Montreal. I was awarded Microsoft Diversity Scholarship in December 2020. Mila (Quebec Artificial Intelligence Institute) provided me with computer equipment and resources to carry out the various experiments.

Chapter 1

Background

Federated Learning is a machine learning paradigm that learns decentralized data thereby preserving user data privacy. Since its conception in 2016 [22], the field has grown exponentially with research and applied contributions from academia and industry. In this chapter, we first motivate federated learning from a privacy perspective as well environmental perspective. We describe the various practical problems of this field. We also introduce a variety of federated settings or types. We further discuss some related federated algorithms for generalization and personalization objectives.

1.1. Introduction to Federated Learning

In today's world, everyone has access to at least one personal phone. Most people have personal laptops or tablets and most homes are equipped with one or more smart assistants and automation setups. With this comes a bulk of informative data. However, with more data regulations like the European General Data Protection Regulation(GDPR) and American Data Privacy and Protection Act in place, collection, transmission, and storage of private data at central servers will not be possible. This affects the current ML and DL methods since they require data available at a central location. Federated learning was introduced by Google to tap this bulk data decentralized across edge devices. It aims to learn the data distributed across devices while preserving privacy of the users since the data remains at the devices where they are generated or collected. To further motivate federated learning, as per a study by Cambridge University on carbon footprint of various ML algorithms, training an FL model emits only about one-tenth of carbon dioxide compared to centralized ML[**43**].

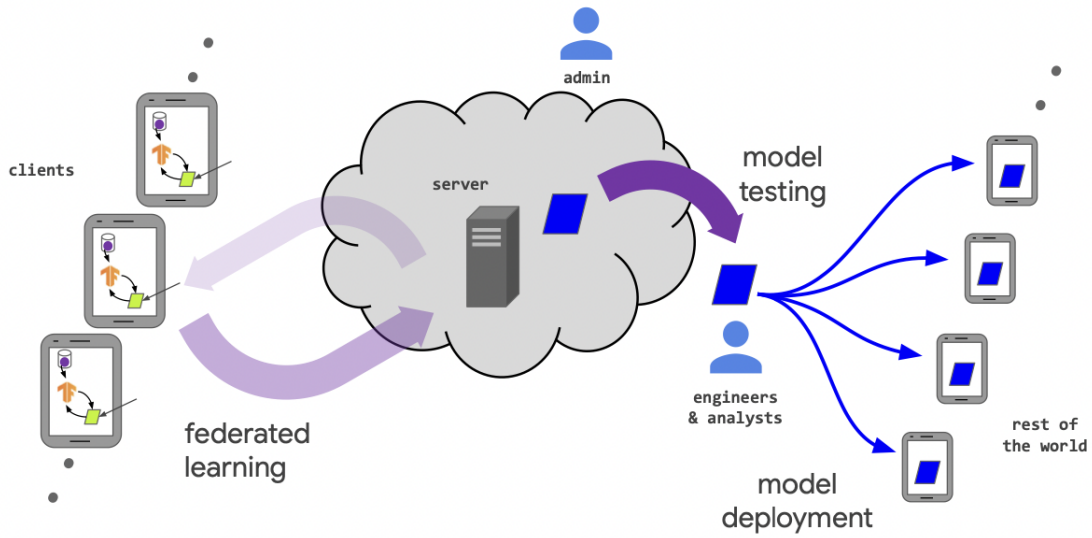


Fig. 1.1. Cross-Device Federated Learning[**25**]

In FL, the clients train on their private data for a few epochs and the client models are then aggregated at the server. The aggregated model then replaces the client models and the process continues until convergence. This ensures data privacy to some extend since the data do not leave the client devices. The only data that is shared from the devices to the server is the client model updates. These updates do not contain more information than the raw data. DP noise can be added to the updates to further enhance privacy.

Though federated learning is of major privacy benefits, it is not devoid of problems. Some of the most important problems are listed below:

- **Data Heterogeneity** The data at the clients are often dependent on the user characteristics and behaviour in case of personal devices. When the clients are hospitals or banks or other silos, the data is often based on factors like geographical location, ethnicity, language and so on. Therefore the data distribution varies from one client to the other. In other words, the data distribution across clients is heterogeneous or non-IID. This gravely affects the model performance since ML assumes that the data is IID.
- **Unbalanced Data** The data generated or collected at the clients would be extremely dependent on the user's device usage patterns in case of edge devices or on the area coverage or subject count in case of institutions. This induces a quantity skew in the data which drastically affects the generalization capability of the FL model.

- **Communication Bottleneck** FL training involves uploading and downloading of client models. Depending on the model size this can be vary large. Furthermore, not all devices would be connected to the network all the time. Due to various issues like lack of connectivity or lack of power, the devices may fail to upload or download the updates. This may slow down model training.
- **Massive network** The number of devices in the federated network can be large. Often the number of data samples per client would be lesser than the total number of clients in the network.

1.2. Federated Learning Settings

Depending on the type of clients involved in the network, federated learning can be cross-device or cross-silo federated learning. Depending on model aggregation it can be split learning or split federated learning. FL without a central server is called decentralized federated learning. When the data is distributed vertically or feature wise across devices, its called vertical federated learning. In this section we give an overview of the various federated settings.

- **Cross-Device FL** When the clients are all edge devices like mobile phones and IoT devices and the number of clients in the federated network is large with full or partial participation in training, it is considered to be cross-device FL. The data at each device is user-specific and extremely heterogeneous. Additionally, the data partition is expected to be horizontal, that is by data samples. Figure 1.1 is a representation of cross-device FL.
- **Cross-Silo FL** When the clients are institutions like hospitals or banks that hold data from multiple individuals, it is considered to be cross-silo FL. The data is skewed in terms of features like geographical location and ethnicity that may be more related to the institution itself than individual users. Additionally, the number of clients in the network is significantly lesser.
- **Decentralized FL** The entire federated learning process is under the orchestration of a server, though it does not have control over data. This makes the server a single point of failure. It may also spread adversarial updates to all clients which may slowly make them corrupted. Decentralized FL or server-less FL completely removes the server from FL using various methods like consensus technique by sharing model updates [**19**] or tree like aggregator structure involving intermediate aggregators [**18**].

- **Vertical FL** When the data is distributed vertically across clients, it is considered to be vertical FL. This is a practically promising area since it allows non-competing institutions or entities to collaborate and boost their performance while not giving up user data privacy [**61**].
- **Split Learning** In split learning, the model is split into two where the input part resides at every client and output part resides at the server. Each client forward passes the input and the representation is sent to the server to continue the forward pass. The gradients are during back propagation is sent back to client with which the client part of the model is updated [**54**].

1.3. Related Works

This section is a brief introduction to various active areas of federated learning research. We also briefly discuss the most common federated algorithms.

In FedAVG [**36**] for each communication round, all selected B fraction of clients perform E local steps of gradient descent with their local datasets. The model parameters from participating clients are averaged at the server to obtain the global model. It is equivalent to FedSGD [**36**] when $E = 1$ and each client performs stochastic gradient descent. Multiple local steps help minimize communication costs, which is a major bottleneck in FL. Quantization methods [**46**] and gradient descent acceleration [**67**] methods have been proposed to reduce communication overhead.

Convergence of FedAVG under i.i.d settings have been analyzed widely [**50, 66, 57**]. The convergence rate of FedAVG worsens with increasing heterogeneity among client datasets and this has been analyzed by several works [**34, 59, 34**]. Multiple variations of FedAVG have been proposed to improve convergence in non-i.i.d data distribution settings, including adding regularization to the client objective [**34**], normalized averaging of model parameters [**58**], and introducing server momentum [**24**]. [**26**] uses control variates to reduce client drift. [**35**] introduces a proximal term at the client loss functions to limit this divergence of the client models by keeping the client model close to the global model. Adaptive optimizers like Adam and Yogi have been introduced to the federated setting by [**45**]. Algorithms like PerFedAVG [**15**], Ditto [**33**], FedBABU [**39**] focus on personalization of clients. Differential privacy and blockchain have been used in FL to enhance data privacy in federated learning [**60, 38**]. Probabilistic Federated Neural Matching (PFNM) [**69**] and FedMA[**56**] addresses the problems due to permutation variances in the neural networks.

Reduction of communication bandwidth required [**36, 21, 47**] and fairness [**35, 37**] in FL are active areas of research. Federated protocols [**3**] and privacy and security [**6, 1**] in FL are other important research areas. Knowledge distillation has been used in FL to improve model performance [**65, 22**]. FedHE[**11**] uses knowledge distillation to learn from models when they are different in architecture. Furthermore, FL has a variety of applications in the real world including but not limited to predictive health models [**8**], communication between vehicles [**48**], learning words [**12**], and next-word prediction [**23**].

Chapter 2

OOD Generalization and Federated Learning

In this chapter we introduce out-of-distribution generalization. We also introduce a few related algorithms that were proposed with the aim of improving OOD generalization performance of the model.We then draw the connections between OOD generalization and federated learning.

2.1. Out-of-Distribution Generalization

In traditional machine learning, a model is evaluated based on its test performance on an unseen dataset drawn i.i.d from the train data distribution. However, this assumption may not hold true in real-world datasets and many supervised learning models do not perform well on related but non-i.i.d test datasets. This problem is often refered to as the out-of-distribution generalization or the closely related domain generalization problem [**27, 2**]. This is because the data used to train models often hold several biases and spurious correlations and models that are trained by minimizing the error on this data inherits those biases and correlations [**2**]. However, to effectively generalize to data from varying distribution, the models should focus on the causal features that are not related to the spurious correlations. These are also called invariant mechanisms and the spurious correlations are often referred to as spurious mechanisms [**2**]. Invariant mechanisms are expected to be prevalent in all data distributions while we do not expect spurious mechanisms to be present in the test distributions or future data distributions. According to [**40**], invariant mechanisms are shared across all environments and are hard to model while spurious mechanisms are easy to spot but are unreliable and varies across environments. An algorithm that generalizes to out-of-distribution data learns the invariant mechanisms while ignoring the spurious correlations.

The OOD generalization problem has been addressed in several works like Invariant Risk Minimization(IRM) [**5**], Risk Extrapolation(REx) [**30**], and Gradient Starvation [**42**]. These approaches typically focus on introducing penalties that learn invariant representations in a setting with known variations in the data (corresponding to environments). [**29**] frames the setting as a game where the algorithm aims to achieve a Nash equilibrium between environments. However, these idea cannot be easily ported to a FL setting as the clients performing the optimization steps would require access to the data of other clients. On the other hand [**40**] proposed a gradient agreement method based on gradient directions to learn features that agree across environments. This was extended by [**49**] to include gradient magnitude. In this paper we focus on this class of gradient agreement methods. Distinct from the prior work, which considers the case of individual samples and single global updates, we consider and adapt this approach to a federated setting, where each client produces an aggregate update based on multiple gradient iterations.

OOD generalization has been explored in [**68**] from a out-of-sample gap or participation gap perspective in federated learning. FL Games[**20**] tries to attain domain generalization by extending IRM Games[**29**].

CausalFed[**17**] explores Out-of-Distribution generalization in a split federated learning setting. This method proposes a federated version of IRM for invariant learning in federated learning. Though the data is decentralized, they use model decoupling. The extractors are at the clients and forward passed in parallel while the representations are shared to the sever classifier where training is completed. This is different from cross-device FL where the models are trained in parallel at the clients and they are aggregated at the server.

In CasalFed[**17**], the participating clients performs local forward passes of the data to extract features in the form of numerical vectors. Consider client data $\mathcal{D}_C = (x_i^C, y_i^C)_{i=1}^{N_C}$ where x_i^C is i^{th} input and y_i^C is i^{th} label for client C. The hidden representation of each participating client is produced as $h_i^C = \phi^C(x_i^C)$; where $h^C \in \mathcal{R}^{N_C \times d}$ and d is the dimension of hidden representation layer. The participating clients sends intermediate client data representations to the server where they are aggregated and trains the server part of the models by minimizing the loss as well as regularizing the model by the gradient norm of the loss for all the participating clients as $\sum_{C,i}^{S,N_C} \mathcal{L}_d(w \circ h_i, y_i) + \lambda \sum_C^S \left\| \nabla_{w|w=1.0} \sum_i^{N_C} \mathcal{L}_d(w \circ h_i, y_i) \right\|^2$, where S equals set of clients, N_C equals number of samples per client C, $\mathcal{L}_d$ equals classification loss, and h, y to represent the hidden representation and its corresponding true class label and λ is hyperparameter. With Invariant Risk Minimization (IRM) [**5**] at the server we attempt to learn invariant predictors in a federated learning setup that can attain an optimal

empirical risk on all the participating client domains. Empirically, CausalFed was observed to improve the OOD generalization performance of models that learn from decentralized data on a variety of datasets including Colored MNIST, Rotated MNIST, and Rotated FMNIST.

2.2. AND-Mask for OOD Generalization

Parascadalo et. al [**40**] hypothesizes that gradient descent leads to loss of information which is potentially important for generalization to out-of-distribution data. This is because gradient descent leads to averaging of loss surfaces, which may converges to a loss mean to all environments or data distributions[**40**]. However this is a "patchwork" solution and may not be invariant across environments. This is because Averaging maximize learning speed or convergence speed and training is often stopped when the loss is low enough. Models learn and converge to the spurious mechanisms in the environments with sole focus on loss minimization, such that the invariances are ignored. According to [**40**], to capture the invariances, learning focus should be consistency. They hypothesize that to generalize to out-of-distribution data, the convergence should be towards a loss that is consistent across environments. This is further intuitively supported by the principle "good explanations are hard to vary", proposed by physiscist David Deutsch[**14**].

The paper [**40**] equates arithmetic mean of loss surfaces to their "Logical OR". It fails to capture the conflicting geometries of landscapes as it performs "Logical OR" on the dominant eigen directions. As an alternative to that, Parascadalo et. al [**40**] propose using geometric mean or "Logical AND" to capture the loss landscapes. These focus on the invariances across landscapes. However, geometric mean cannot be directly applied due to several practical limitations like instability induced by the presence of zero gradients, strict requirement for all gradients to be of the same sign, and log domain computations.

Handling the above problems, Parascadalo et. al [**40**] propose using a practical binary mask that zero out gradients with inconsistent signs. They call it an AND-Mask. The gradients are masked as $m_\tau * \nabla L_e$; where m_τ zeros out gradients that have less than τ consistent gradients across environments.

$$[m_\tau]_j = \mathbb{1}\|\frac{1}{|N|}\sum_{e\in\eta}\text{sign}([\nabla L_e]_j) \geq \tau\|$$

Here ∇L_e is the gradient of the loss with respect to environment e, and $\tau \in [0, 1]$ is a hyper-parameter.

2.3. Connections between OOD Generalization and FL

OOD generalization in traditional machine learning involves centralized data and is often formalized using the notion of domains or environments. Under the formalism of [**5**] an environment corresponds to a data generating distribution that can be related through underlying (potentially unknown) causal variables to a set of other environments. Different environments can arise during model training and testing, while it is typically assumed all environments (train and test) share some invariant mechanisms. They can however have spurious mechanisms that differ across environments [**40, 9**]. The concept of environments can be related to the federated learning setting involving decentralized data by considering each client as producing a set of data generated from a different environment. All clients have underlying invariant mechanisms to be considered for training a global model. Each client also has their specific spurious mechanisms or data distributions. For example, consider a scenario of different clients corresponding to smartphone users capturing pictures of fruits to build a model that identifies fruit items. Each smartphone may have different camera and each user may take pictures of different subsets of fruit. Thus the clients may differ in terms of the label distribution of their local data and the camera related image characteristics, which can be a spurious mechanism while the overall set of food items is invariant across clients in the federated network.

The objective of OOD generalization is to improve the performance of a model on data from distributions that are related yet different from the training data distribution. [**4**] quantifies the above-mentioned objective as $R^{OOD}(f) = \max_{e \in \xi_{all}} R^e(f)$ where $R^e(f)$ is the risk or expected loss for data from environment e, which belongs to ξ_{all}, a large (often infinite) family of distinct yet related environments. In practical FL, one of the major objectives of the global model is to improve its performance on non-participating clients (clients that do not contribute to global model training [**25**]) and on new train clients (participating clients that are new to the federated network). The data at these clients will be from related distributions having the same invariant mechanism but may differ from data distributions at train clients. Hence, one way to frame the goal for FL global models is to enhance the performance across a large set of related clients which may have different data distributions. We can quantify this as $\min R_{gFL}(f) = \max_{c \in C_{all}} R^c(f)$ where $R^c(f)$ is the risk at client $c \in C_{all}$, and C_{all} is a family of probability distributions, with c corresponding to the distribution of a

unique client in a FL framework.

Chapter 3

Gradient Masked Averaging

In this chapter we describe the proposed algorithm in depth. We start with an introduction of the AND-Mask introduced in [**40**] for OOD generalization in a centralized setting. We then describe the motivations for using this algorithm to increase global model generalization performance in federated learning followed by its shortcomings. Further, we describe the proposed algorithm in detail. We also provide theoretical guarantees to support the algorithm.

3.1. Gradient Masked Averaging

Inspired by AND-Mask and the Connections between OOD-generalization and FL (Chapter 2) we propose Gradient Masked Averaging for generalization in federated learning. In this chapter we first discuss the motivations for using AND-Mask, followed by details on transition from the binary AND-Mask to a Soft Mask. Then we discuss the full algorithm in detail.

3.1.1. Motivation

AND-Mask introduces τ as a hyper-parameter to threshold the agreement across environments. To understand the relation between the effect of agreement threshold and the consistency of gradients across clients, we manually vary the level of heterogeneity and record the fraction of clients that have gradient agreement less than $\tau = 0.4$ in Figure 3.1. The heterogeneity was induced using a Dirichlet distribution based label skew on MNIST with $\alpha = 0.1$ representing heterogeneous case and $\alpha = 100$ representing homogeneous case. We observe that as homogeneity increases or as clients become more alike each other, the

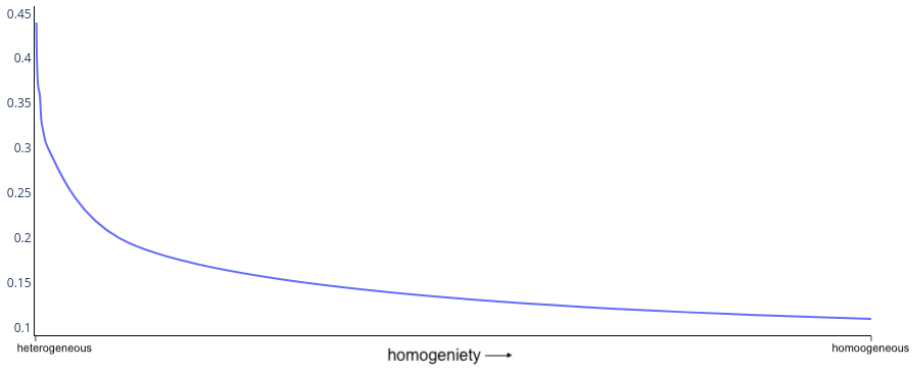


Fig. 3.1. Fraction of clients having *agreement* $< \tau (= 0.4)$ vs. homogeneity of data

fraction of clients having gradient agreement less than tau decreased. In other words, with increasing homogeneity, the number of gradients having agreement greater than tau increased. This implies a connection between gradient agreement across clients and data heterogeneity. Therefore, agreement threshold τ can be used in federated learning to implicitly identify the heterogeneity in data.

3.1.2. Binary AND-Mask to Soft Mask

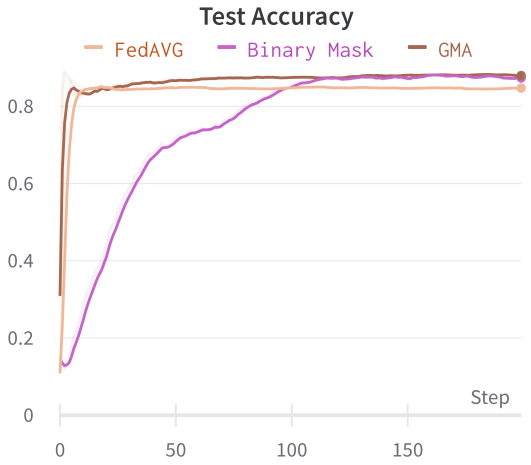


Fig. 3.2. FedAVG[**36**] vs. FedGMA vs. Binary Mask [**40**] on MNIST with label skew.

Parascadalo et. al [**40**] propose using a binary mask to zero out incosistent gradients. Direct application of this idea to FL setting is however challenging. [**40**] applies the rule assuming each sample represents an environment, whereas each client more naturally corresponds to the environment in FL. Furthermore, they show that this can lead to a slower convergence rate in practice as too many components can be masked at each iteration. The same was observed on using the binary mask in a federated setting. As expected from the results of Parascandalo et. al. [**40**], the binary mask converges to a higher test accuracy than averaging, which corresponds to FedAVG in federated learning. However, the number of communication rounds required to achieve it was higher when the data was non-iid as shown in 3.2. The heterogeneity introduced here is label skew on MNIST. In a federated setting, this would be impractical as we would not want to sacrifice convergence speed for generalization. Communication bottleneck is severe practical federated learning settings.

3.1.3. Algorithm

Algorithm 1 Gradient Masked FedAVG [**36**]

Server Executes:
Initialize w_0
for each server epoch, t = 1,2,3,... **do**
Choose C clients at random
for each client in C, n **do**
$w_t^n = \text{ClientUpdate}(w_{t-1})$
$\Delta_t^n = \frac{n_s}{\sum_{n \in C} n_s}(w_t^n - w_{t-1})$
end for
$\Delta_t = \sum_{n \in C} \Delta_t^n$
$b = \tilde{m}_\tau(\{\Delta_t^n\}_{n \in C})$
$w_t = w_{t-1} + \eta_g * b \odot \Delta_t$
end for

ClientUpdate(w):
Initialize $w_0 = w$
for each local client iteration, i=0,1,2,3,..,n **do**
$g_i = \nabla_{w_i} L(w_i)$
$w_{i+1} = w_i - \eta_c \, g_i$
end for
return w_{i+1} to server

Algorithm 2 Gradient Masked, FedADAM, and FedYogi [**45**]

Server Executes:

Initialize w_0
for each server epoch, t = 1,2,3,... **do**
Choose C clients at random
for each client in C, n **do**
$w_t^n = \text{ClientUpdate}(w_{t-1})$
$\Delta_t^n = \frac{n_k}{\sum_{n=1}^{N} n_k}(w_t^n - w_{t-1})$
end for
$\Delta_t = \sum_{n=1}^{N} \Delta_t^n$
$z_t = \beta_1 z_{t-1} + (1-\beta_1)\Delta_t$
$v_t = v_{t-1} - (1-\beta_2)\Delta_t^2 \text{sign}(v_{t-1} - \Delta_t^2)$
$v_t = \beta_2 v_{t-1} + (1-\beta_2)\Delta_t^2$
$\Delta_t = \frac{z_t}{\sqrt{v_t}+e^{-3}}$
$b = \tilde{m}_\tau(\{\Delta_t^n\}_{n=1..C})$
$w_t = w_{t-1} - \eta_g * b \odot \Delta_t$
end for

In the federated setting, we propose a variant of the binary mask that doesn't sacrifice convergence speed while retaining some of the improved generalization properties. Specifically, we propose to use masking at the aggregation stage of standard FL, with a mask computed based on each client update (which arise from multiple local gradient steps). The mask is calculated based on sign agreement among client updates Δ_n and it is applied on the global model update Δ_n^k. This masking controls the parameter update based on the agreement of direction among the gradients across clients or environments. To provide rapid convergence we apply a soft masking procedure instead of the hard binary mask. In Figure 3.2, we observe that the proposed algorithm (marked as GMA) which use the soft-mask is capable of converging to the higher accuracy achieved by binary mask, while not requiring as many communication rounds. Number of communication rounds required for GMA to converge is almost equal to that required by FedAVG.

We define an agreement score, $A_j \in (0,1]$, given as a function of all the client updates and mask $\tilde{m}_\tau$ is defined element-wise,

$$[\tilde{m}_\tau]_j = 1 \textbf{ if } \mathrm{A}_j \geq \tau \textbf{ else } \mathrm{A}_j \ \textit{where} \ \mathrm{A} = \left| \frac{1}{|N|} \sum_{n \in N} \text{sign}(\Delta^n) \right| \tag{3.1.1}$$

Algorithm 3 Gradient Masked SCAFFOLD [**26**]

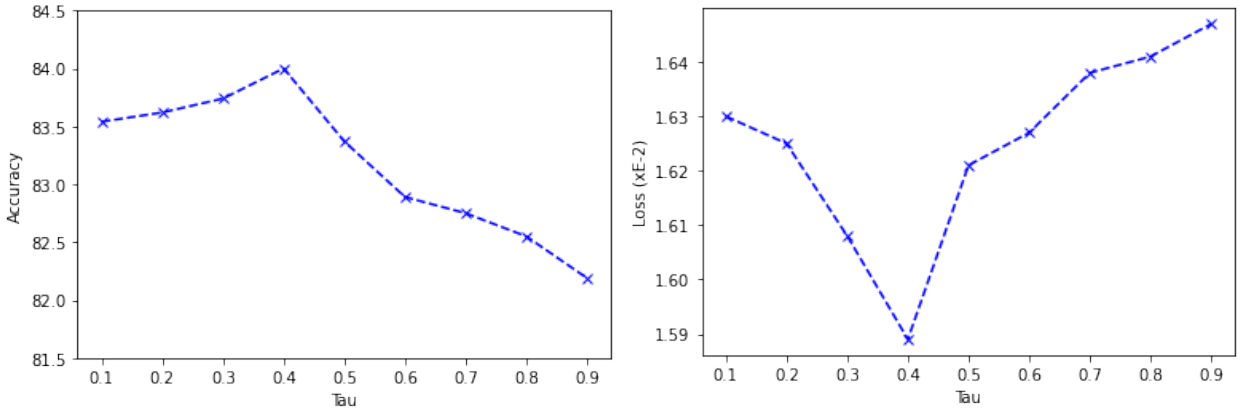
Server Executes:

Initialize w_0
for each server epoch, t = 1,2,3,... **do**
  Choose C clients at random
  for each client in C, n **do**
    $w_t^n = \text{ClientUpdate}(w_{t-1})$
    $w_t^n, \Delta_c^n = \text{ClientUpdate}(w_{t-1}, \Delta_c)$
    $\Delta_t^n = \frac{n_k}{\sum_{n=1}^{N} n_k}(w_t^n - w_{t-1})$
  end for
  $\Delta_t = \sum_{n=1}^{N} \Delta_t^n$
  $\Delta_c = \frac{1}{N} \sum_{n=1}^{N} \Delta_c^n$
  $mask = \tilde{m}_\tau(\{\Delta_t^n\}_{n=1..C})$
  $w_t = w_{t-1} - \eta_g * mask \odot \Delta_t$
end for

ClientUpdate(w):

Initialize $w_0 = w$
$c_i = c_i^+$
for each local client epoch, i=0,1,2,3,..,n **do**
  $g_i = \nabla_{w_i} L(w_i)$
  $w_{i+1} = w_i - \eta_c \; g_i - c_i + c$
end for
$c_i^+ = (i) g_i(x) \text{or } (ii) c_i - c + \frac{1}{K\eta_l}(x - y_i)$
return w_{i+1} , $c_i^+ - c_i$ to server

The global model update is given by $\tilde{m}_\tau \odot \Delta_t$. This ensures that the updates to the global model are with respect to their agreement across clients. When the agreement across clients is greater than the hyperparameter τ, it would be assigned 1 and when the agreement is lesser than τ, the mask value would be equivalent to the agreement score. This real mask ensures that each parameter updates but the magnitude is adjusted to be proportional to the agreement across clients. Furthermore, we observe empirically in Figure 3.1 that the fraction of clients whose magnitude gets adjusted as mentioned above is correlated to the heterogeneity in the distribution of data across clients. The full algorithm for Gradient Masked Aggregation on FedAVG is given in Algorithm. 1. Extended version of GMA on FedAdam and FedYogi are given in Algorithm 2 and GMA on SCAFFOLD is given in Algorithm 3. Client updates in Algorithm 2 is same as that in Algorithm 1



(a) Test accuracy vs. τ **(b)** Test loss vs. τ

Fig. 3.3. Effects of τ on model performance

3.1.4. Effect of τ

$\tau \in \{0,1\}$ is a hyper-parameter introduced to threshold agreement across clients. It marks the minimum agreement required to consider the gradient for aggregation. When $\tau = 0$ or negligible, the gradients of all parameters will have an agreement score greater than or equal to τ. This makes all gradients consistent across clients. The agreement score would be over-written by 1 and the equation becomes equal to that of naive federated aggregation where all gradients are averaged with the same importance. This is equivalent to an underfit condition. The opposite overfit criterion can happen with a high τ value. In this case, no gradient would be considered dominant and all parameters updates would be diminished corresponding to their agreement score. The agreement score can be 1 when the data distribution across clients is an ideal i.i.d distribution where all gradients across clients would be along the same direction. But in practical scenario, such data distributions are rare in a federated setting. When agreement $= 1$, $w^k = w^{k-1} - \eta_g \, 1 \odot \Delta^k = w^k - \eta_g \Delta^k$; equivalent to naive federated aggregation. This implies that the naive federated aggregation is a case of the proposed gradient masked averaging.

$\tau = 0.4$ implies that the gradient being considered have a 40% excess or a total of 60% of the client gradients along the dominant direction. From our experiments it was observed that when τ is low, the model underfits. The accuracy was best at $\tau = 0.4$ and on further increasing τ, it was overfitting. This is visible from the test accuracy vs. τ plot in Figure 3.3a and test loss vs. τ plot in Figure 3.3b .

3.2. Federated Aggregation Preliminaries

In this section we introduce the notations used and review standard federated aggregation. Consider a federated setting having N clients where the data at each client, $n \in N$ is $D^n = (x_i^n, y_i^n)$ and each client have n_s samples. The clients collectively learn a function, $f : X \to Y$ $f(x \in X; w)$ that, in our case, corresponds to a neural network model with parameters, w, and $(X,Y) = \{(x_n,y_n) : \forall n \in N\}$ is the entire set of data distributed across clients. At each communication round, k, the parameters of the global model, w_k, are sent to participating clients who perform multiple local gradient steps to obtain an update, Δ_n^k, corresponding to the difference between the clients model after multiple updates and w_k. In most FL algorithms the global model update at k^{th} communication round is then obtained as

$$w^{k+1} = w^k - \eta_g \Delta^k \quad where \quad \Delta^k = \frac{1}{|N|} \sum_{n \in N} \Delta_n^k \tag{3.2.1}$$

η_g is the global learning rate and for sufficiently large K, $\eta_g = K\eta_l$ [**26**]. Δ^k is the update or "pseudo-gradient" at the k^{th} global communication round obtained by aggregating the updates from the participating clients (Δ_n^k; $n \forall N$). The pseudo-gradient is an approximation of the global model gradient which is used for the model update.

In the case of a single gradient step at each client, the update Δ^k, corresponds to the gradient of the global objective. Each client has a different data distribution and thus different loss surface. [**40**] shows that averaging of gradients across environments leads to poor consistency of solutions, and reduced generalization, particularly to unseen environments. Indeed naive averaging of parameters fails to capture the consistencies in the loss landscapes due to the bias that may be induced by dominant features in the environments as explained by [**49**]. This is further exacerbated in real world federated settings as there are multiple possible scenarios where some clients dominate over others.

3.3. Theoretical Guarantees

We now analyze the convergence properties of the masking, focusing on the case of FedAVG and gradient masked aggregation. In our setting, we define the global objective function in terms of the local objective, F_n, of each client as shown below, assuming that

$\mathbb{E}[f_n(w)] = F_n(w)$

$$\min_w f(w) = \min_w \sum_{n=1}^{N} F_n(w)$$

Following [**45**], we make the following standard assumptions. We write $\mathcal{F}_t$ the filtration adapted over our stochastic process at time t.

Assumption 3.3.1 (Lipschitz gradient)**.** We assume that each client objective has Lipschitz gradient with constant L, meaning that there exists $L > 0$, $\forall n, \forall w,v$, $\|\nabla F_n(w) - \nabla F_n(v)\| \leq L\|w - v\|$

Assumption 3.3.2 (Bounded gradients)**.** We assume that each client has a bounded gradient by G, leading to: $\exists G > 0, \forall n, \forall w, \|\nabla F_n(w)\| \leq G$.

Assumption 3.3.3 (Finite variance)**.** We assume a global bound on the variance of the gradient estimate of each individual client, meaning that: $\exists \sigma > 0, \forall n, \forall w, \mathbb{E}\|\nabla F_n(w) - \nabla f_n(w)\|^2 \leq \sigma^2$.

Assumption 3.3.4 (Global variance)**.** We assume a global bound on the variance of the gradient estimate of each individual client, meaning that: $\exists \sigma_g > 0, \forall n, \forall w, \mathbb{E}\|\nabla F_n(w) - \nabla f(w)\|^2 \leq \sigma_g^2$.

Lemma 3.3.5 (Bounded drift from client update, Adapted from Appendix A, Lemma 3 of [**45**])**.**

Given the above Assumptions, there exists $C > 0$ such that for any time step t and x_t,Δ_t obtained from Alg. 1:

$$\mathbb{E}[\|\Delta_t - \nabla f(w_t)\|^2] \leq C(\sigma^2 + \sigma_g^2 + \mathbb{E}[\|\nabla f(w_t)\|^2])$$

Démonstration. Using the Appendix A, Lemma 3 of [**45**] is exactlying saying, assuming the number of iterations performed by a client is bounded (and for local step sizes defined in [**45**]), that:

$$\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}\|w_t^n - w_{t-1}\|^2 \leq C(\sigma^2 + \sigma_g^2 + \mathbb{E}\|\nabla f(w_t)\|^2)\,,$$

for some absolute constant $C > 0$. and now we observe that:

$$\mathbb{E}\|\Delta_t\|^2 = \mathbb{E}\|\sum_{n=1}^{N} \Delta_t^n\|^2 \tag{3.3.1}$$

$$\leq N \sum_{n=1}^{N} \mathbb{E}\|w_t^n - w_{t-1}\|^2 \tag{3.3.2}$$

$$\leq N^2 C(\sigma^2 + \sigma_g^2 + \mathbb{E}\|\nabla f(w_t)\|^2) \tag{3.3.3}$$

Now, we know that:

$$\mathbb{E}[\|\Delta_t - \nabla f(w_t)\|^2] \leq 2\mathbb{E}[\|\Delta_t\|^2 + 2\mathbb{E}[\|\nabla f(w_t)\|^2]$$

and we get the conclusion. □

The above inequality can be deducted from the aforementioned paper by using the L-smoothness of f_n(as done (Eq. 6) in [**45**]), with K local steps and local parameters $w_{t,n,k}$ on client n from time t. This Lemma involves the aggregation at every step t of the local client updates obtained individually on each client. In particular, it does not depend on the server's algorithm. Due to this, the proof from Appendix A, Lemma 3 of [**45**], which gives the explicit C, applies directly.

The next proposition derives a rate of convergence on the masked gradient which is similar to [**45**], and in the order of $\mathcal{O}(\frac{1}{T})$.

Proposition 3.3.6 (Convergence analysis)**.** *Given assumptions, if* $0 \leq \eta_g \leq \frac{1}{2L}$ *, then, one has the following rate over the masked gradients given by the FedAVG algorithm in :*

$$\mathbb{E}[\min_{t<T} \|b \odot \nabla f(w_t)\|^2] \leq 2L \left(\frac{f(w_0) - f(w_T)}{T} + C\eta_g \left(\sigma^2 + G^2 + \sigma_g^2\right) \right)$$

Démonstration. We consider the optimization path given by Alg 1. Let us write $\tilde{\Delta}_t = b_t \odot \Delta_t$. First, we note that given that $0 \leq b_t^j \leq 1$, we get $\|\tilde{\Delta}_t\| \leq \|\Delta_t\|$. Next we follow the approach of [**7**] for obtaining optimal non-convex bounds. Each f_n is L-smooth, thus:

$$\begin{aligned} F_n(w_{t+1}) &\leq F_n(w_t) + \langle \nabla F_n(w_t), w_{t+1} - w_t \rangle + \frac{L}{2}\|w_{t+1} - w_t\|^2 \\ &= F_n(w_t) - \eta_g \langle \nabla F_n(w_t), b_t \odot \Delta_t \rangle + \frac{L}{2}\|b_t \odot \Delta_t\|^2 \\ &\leq F_n(w_t) - \eta_g \langle \nabla F_n(w_t), b \odot \Delta_t \rangle + \frac{L}{2}\eta_g^2 \|b_t \odot \Delta_t\|^2 \end{aligned}$$

Averaging over $1 \leq n \leq N$ and conditioning over $\mathcal{F}_t$ leads to:

$$\begin{aligned}\mathbb{E}[f(w_{t+1})|\mathcal{F}_t] &\leq f(w_t) - \eta_g \langle \nabla f(w_t), b_t \odot \Delta_t \rangle + \frac{L}{2}\eta_g^2 \|b_t \odot \Delta_t\|^2 \\ &= f(w_t) - \eta_g \langle \nabla f(w_t), b_t \odot (\nabla f(w_t) - \nabla f(w_t) + \Delta_t) \rangle + \frac{L}{2}\eta_g^2 \|b_t \odot \Delta_t\|^2\end{aligned}$$

Now, we use the inequality: $\langle a,b \rangle \leq \|a\|\|b\| \leq \frac{1}{2}(\|a\|^2 + \|b\|^2)$ and noting the masking can be seen as multiplication by diagonal matrix, we obtain:

$$\begin{aligned}\langle \nabla f(w_t), b \odot (\nabla f(w_t) - \Delta_t) \rangle &= \langle b \odot \nabla f(w_t), \nabla f(w_t) - \Delta_t \rangle \\ &\leq \frac{1}{2}\|b \odot \nabla f(w_t)\|^2 + \frac{1}{2}\|\nabla f(w_t) - \Delta_t\|^2 .\end{aligned}$$

From the Bounded gradients and Lemma **??**, we get:

$$\mathbb{E}[\|\nabla f(w_t) - \Delta_t\|] \leq C(\sigma^2 + \mathbb{E}[\|\nabla f(w_t)\|^2 + \sigma_g^2) \leq C(\sigma^2 + G^2 + \sigma_g^2)$$

Since $0 \leq b^j \leq 1$, we get:

$$-\nabla f(w_t)^j \times b_t^j \times \nabla f(w_t)^j \leq -(b^j)^2 (\nabla f(w_t)^j)^2 ,$$

which implies that:

$$-\langle \nabla f(w_t), b \odot \nabla f(w_t) \rangle \leq -\|b \odot \nabla f(w_t)\|^2$$

Taking the expectation, and summing, we have:

$$\frac{1}{2}(\eta_g - L\eta_g^2) \sum_{t=0}^{T-1} \mathbb{E}[\|b \odot \nabla f(w_t)\|^2] \leq f(x_0) - f(x_T) + \eta_g T C(\sigma^2 + G^2 + \sigma_g^2)$$

In particular, this implies for a learning rate $\eta_g = \frac{1}{2L}$ small enough such that $\eta_g - L\eta_g^2 = \frac{1}{2L} > 0$

$$\mathbb{E}[\min_{t<T} \|b \odot \nabla f(w_t)\|^2] \leq 2L \left(\frac{f(w_0) - f(w_T)}{T} + C\eta_g \left(\sigma^2 + G^2 + \sigma_g^2 \right) \right)$$

□

We now observe that under assumptions similar to those proposed in [**40**], the distribution of updates will match the true underlying distribution.

Proposition 3.3.7 (Mask stability)**.** *Denote δ,δ^n the random variable corresponding respectively to a coordinate of Δ,Δ^n. Furthermore consider $\tilde{\delta}$ the random variable for each coordinate of $\tilde{\Delta}$, where $\tilde{\Delta} = b \odot \Delta$. Assume that δ^n is σ-sub-Gaussian, that the δ_n are mutually independent and write $\mu^n = \mathbb{E}[\delta^n]$. If $\frac{1}{N}$ `card`$(\{n|\mu^n > 0\}) > \tau$, then, with probability $1 - \mathcal{O}\left(e^{-\frac{(\inf_{\mu_n>0} \mu_n)^2}{\sigma^2}}\right)$, we obtain $\tilde{\delta} = \delta$.*

DÉMONSTRATION. We show a lower bound on δ^n. With probability $1 - e^{-\frac{t^2}{\sigma^2}}$, we get $|\delta^n - \mu^n| < t$. Let's thus pick $\tilde{t} = \inf_{\{\mu^n > 0\}} \frac{\mu^n}{2}$. By considering the intersection of those events, it implies that with probability at least $1 - e^{-\frac{\tilde{t}^2}{\sigma^2}}$, $\delta^n > \mu^n - \frac{\mu^n}{2} = \frac{1}{2}\mu^n > 0$. Consequently, the mask is equal to 1 and $\delta^n = \tilde{\delta}^n$. Now, we can note that $\inf_{\mu_n > 0} \mu_n > \inf_{\mu_n \neq 0} |\mu_n|$, which allows to conclude. □

Informally we see that if w_t is far from a local minimum then the masked gradient $b_t \odot \Delta_t$ is likely to not be equal to 0 thanks to Prop 5.6. Thus, Prop 5.5 suggests that the norm of the gradient is decreasing.

Chapter 4

Experiments

In this chapter, we show empirically that the proposed GMA tends to outperform standard aggregation (AVG), converging at similar or better than standard aggregation, while enhancing the global model generalization. We observe this for multiple FL algorithms with respect to multiple datasets and data distributions (i.i.d and non-i.i.d).

Implementation We conduct experiments on gradient masked and naive versions of non-adaptive federated optimizers like FedAVG [**36**], FedProx [**35**], and SCAFFOLD [**26**] and adaptive optimizers like FedADAM and FedYogi [**45**] across a variety of datasets. Our experiment include label distribution skew, feature distribution skew and quantity skew. The reported performances are average accuracies of 4 independent runs of the model on a test dataset. The implementation was an adaptation from that in [**35**].

Hyperparameters An SGD optimizer with a momentum ($\rho = 0.9$) and cross-entropy loss was used to train each client ($N = 10$ and $C = N$) for $E = 1$ client epochs before aggregation at the server in all our experiments unless specified. For experiments with non-convex objectives, LeNet architecture was employed at all clients and at the global model for all datasets except CIFAR-10, which used a ResNet18 with group norm [**45**]. The momentum parameters of adaptive federated optimizers are fixed at $\beta_1 = 0.9$ and $\beta_2 = 0.99$ as per [**45**]. For each of the considered algorithms we tune the local client model learning rates and global model learning rates to consider the best performances of the algorithms. The global learning rate, client learning rate, and τ are tuned in the range given below.

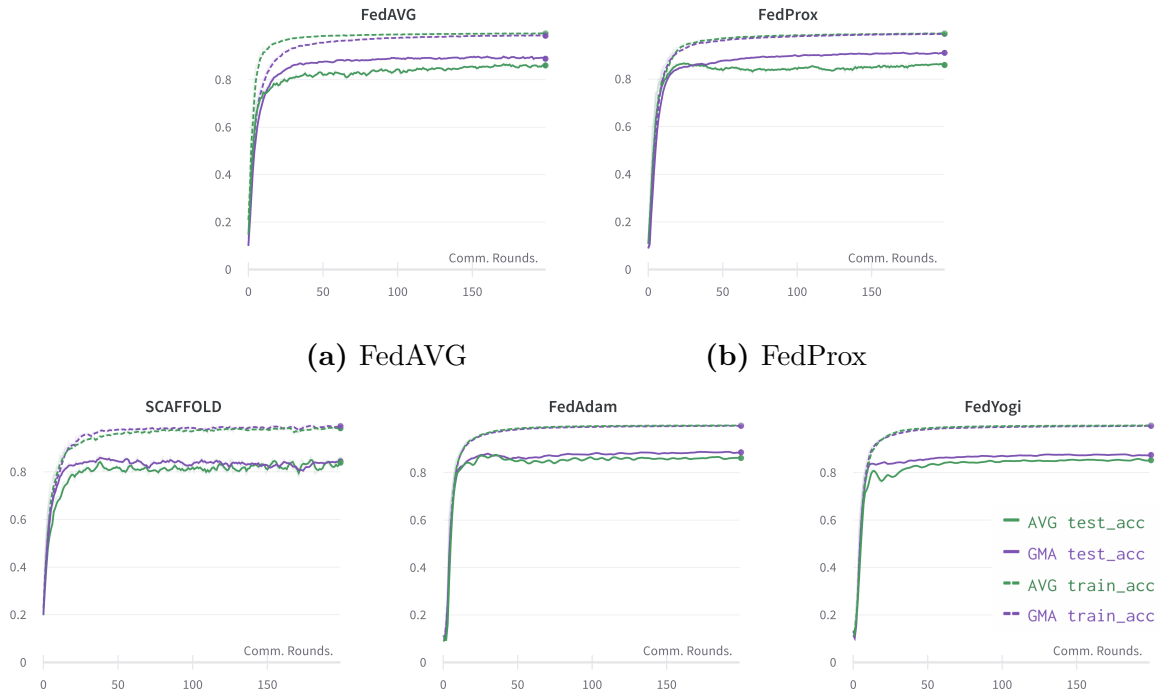
$$\eta_{\mathbf{l}} \in \{10^{-3}, 10^{-2}, 5.10^{-2}, 10^{-1}\}$$
$$\eta_{\mathbf{g}} \in \{10^{-2}, 10^{-1}, 1, 1.5, 2\}$$
$$\tau \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$$

Tableau 4.1. Real-World evaluation on FEMNIST and Out-of-Distribution evaluations with and without label distribution skew on FedCMNIST and FedRotMNIST. Average best test performance(%) over 4 independent runs of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions are reported below. The best average result among AVG and GMA having atleast 0.01% higher than the other algorithm is shown in bold.

Dataset (Model)	Label Skew	FedAVG AVG	FedAVG GMA	FedProx AVG	FedProx GMA	SCAFFOLD AVG	SCAFFOLD GMA	FedAdam AVG	FedAdam GMA	FedYogi AVG	FedYogi GMA
Real-World Evaluation											
FEMNIST (LeNet)	✗	99.31±0.01	**99.38**±0.01	99.24±0.01	**99.33**±0.01	98.88±0.04	**99.36**±0.03	99.12±0.01	**99.16**±0.00	99.14±0.01	**99.18**±0.00
Out-of-Distribution Evaluation											
FedRotMNIST (LeNet)	✗	99.11±0.02	99.11±0.01	99.14±0.02	99.14±0.03	99.09±0.01	**99.10**±0.02	98.71±0.05	**98.8**±0.02	98.73±0.04	**98.78**±0.02
	✓	98.88±0.04	**98.94**±0.01	98.90±0.03	**98.95**±0.01	98.94±0.04	**98.97**±0.03	98.21±0.01	**98.33**±0.02	98.41±0.07	**98.56**±0.04
FedCMNIST (LeNet)	✗	89.37±0.83	**90.36**±0.61	89.61±0.88	**90.22**±0.78	88.14±0.53	**89.54**±0.38	88.79±0.86	**89.4**±0.84	88.78±0.83	**89.88**±0.75
	✓	86.77±0.43	**89.17**±0.38	86.84±0.38	**89.33**±0.34	86.41±0.8	**89.49**±0.63	85.75±0.53	**89.28**±0.44	86.67±0.53	**89.76**±0.37

4.1. Real-World Evaluation

In the practical federated setting, the data across clients is heterogeneous and the clients which deploy the global model (including test clients, non-participating clients, and new clients in the federated network) can have data distribution different from that at any train clients. This can be simulated by using a realistic federated data characterised by a feature distribution skew unique to each client including the test client. Specifically we use train data from the same domain distributed across clients such that each client has data corresponding to one user (or a set of users) unique to the client. The test data consists of data from the same domain as the train dataset distributed across clients but from one user (or a set of users) not included in the set of train clients. A similar feature skew is provided by Federated EMNIST [**10**] where the data points have a user identifier. The data is distributed amongst the clients based on the identifiers in a way that no clients share data corresponding to the same user. The test performance of the algorithms and their gradient masked alternatives are given in Table 4.1. We observe that gradient masking outperforms naive averaging. In the next section we consider a more complex out-of-distribution feature skew to further evaluate GMA.



(a) FedAVG **(b)** FedProx

(c) SCAFFOLD **(d)** FedAdam **(e)** FedYogi

Fig. 4.1. Train accuracy and test accuracy vs. communication rounds of gradient masked and naive averaging versions of the algorithms on FedCMNIST distributed non-i.i.d across clients.

4.2. Out-of-Distribution Evaluation

A more complex OOD test can be implemented to better understand the performance of gradient masking in federated learning. For this we induce unique spurious mechanisms or features in the clients (including test) besides the class label based heterogeneity. The global model would be tested on a dataset having a spurious mechanism that was not present in any of the train clients, while the spurious mechanism at each train client is unique to itself. The label skew is induced such that each client would have 90% samples from two classes and the 10% noise would comprise samples from all other eight classes. Tables 4.2 and 4.3 shows a sample label skewed data distribution across ten clients.

We use FedCMNIST, a federated multiclass version of CMNIST [**5**] with multiple color-label correlations. The invariant mechanism here is the digit. There also exists a spurious mechanism marked by a color given to the numbers. The color is digit specific to induce correlation to the label. Specifically, each digit would have one or more color that remains the same across examples in the train set or in the data at the participating

clients. The color of the same digit in the test set would be different from that at any of the train clients. This spuriousness is in addition to the label distribution based skew with non-iid distribution of data across clients. We also use FedRotMNIST, inspired from [**16**], where an angle of rotation is the spurious mechanism. Specifically, we rotate each digit at an angle such that a label based correlation is induced. In our experiments the digits were rotated at 10, -10, 20, -20, 30, -30, 40, -40, 50, and -50 respectively. The test images are not rotated at any angle irrespective of the digit or label. The preprocessing includes padding on rotation and cropping. The performance of the various algorithms and their gradient masked averaging counterparts on these OOD test datasets is given in Table 4.1. It is to be noted that across all datasets, data distribution, and algorithms gradient masking outperforms naive averaging. Figure 2 shows train and test curves of the algorithms and their gradient masked alternatives on non-i.i.d distribution of FedCMNIST. It is to be noted that although GMA train accuracies are less than or equal to that of naive averaging, the GMA test accuracies are higher. This indicates that gradient masked versions generalize better than the naive versions.

4.3. In-Distribution Evaluation

This is the most widely considered setting in the FL literature. The global model is evaluated on a test dataset sampled from data at all clients irrespective of the data distribution across clients. This test dataset is a representation of all participating clients. The datasets used for in-distribution testing are MNIST [**31**], Fashion MNIST [**63**], FEMNIST (Federated EMNIST) [**10, 13**], and CIFAR-10 [**28**].

Table 4.4 shows the test performance of the algorithms and their GMA versions. It was observed that with the robust features of GMA, the algorithm is capable of outperforming naive averaging. The difference in improvement is more significant when the data distribution is non-i.i.d. The major reason for this is that gradient masking is capable of focusing on learning the invariances even under increased spuriousness of non-i.i.d data distribution. Across datasets, we can observe that the improvement with gradient masking is more prominant on CIFAR10 and FMNIST, which are relatively complex datasets than other datasets considered. Furthermore, Table 4.6 shows an ablation comparing GMA and AVG when the same client and global rates are used, showing that for nearly any hyperparameter choice GMA outperforms AVG, suggesting it is highly robust to the choice of hyperparameters.

Tableau 4.2. This table shows the label distribution across clients for an IID setting. Each client will have randomly chosen examples from all 10 classes. This represent the 10 class setting in MNIST. We have considered 3 clients for the table. The same pattern would be present across all clients.

	0	1	2	3	4	5	6	7	8	9
Client 1	585	643	591	550	571	561	631	628	620	620
Client 2	589	691	593	628	553	526	588	640	602	590
Client 3	531	697	595	627	557	557	596	626	581	633
.....										

Tableau 4.3. This table shows the label distribution skew used by us for our experiments for a non-IID data distribution across clients. This represents the 10 class setting in MNIST. We have taken 3 clients. The same pattern would be present across all clients.

	0	1	2	3	4	5	6	7	8	9
Client 1	2894	2247	51	48	50	53	52	47	47	47
Client 2	44	2246	1962	40	42	42	42	41	41	46
Client 3	31	31	33	1962	33	1371	35	31	31	32
.....										

Using the non-iid distributed FMNIST data we further study how the performance is affected as the number of clients grows (N = 10,50,100,250) and the number of local epochs increases (E = 1,3,5,10,20). The results are shown in Figure 4.2a and Figure 4.2b. It can be observed that gradient masking increasingly outperforms naive averaging in these more complex scenarios. In Figure 4.2a we observe that with increasing number of clients, difference between the test accuracies corresponding to GMA and naive averaging increases. This validates the enhanced invulnerability of gradient masking to the bias that could be induced by one or more clients in the network. In Figure 4.2b we observe increasing local epochs beyond 3, the test accuracy decreases due to client drift [**26**]. Gradient masking is however more robust in this (challenging) scenario.

To further understand the performance of the proposed algorithm under larger federation, we experiment on CIFAR10 and CIFAR100 with larger number of clients and with random sampling of clients per communication round for training. We randomly sample $C = 10$ clients per round from a total of $N = 500$ or $N = 100$ clients in the federated network

Tableau 4.4. In-Distribution evaluations with and without label distribution skew on MNIST, FMNIST, FEMNIST, and CIFAR10 and Quantity skew based evaluation on CIFAR10. Average best test performance(%) over 4 independent runs of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions are reported below. The best average result among AVG and GMA having atleast 0.01% higher than the other algorithm is shown in bold.

Dataset (Model)	Label Skew	FedAVG AVG	FedAVG GMA	FedProx AVG	FedProx GMA	SCAFFOLD AVG	SCAFFOLD GMA	FedAdam AVG	FedAdam GMA	FedYogi AVG	FedYogi GMA
MNIST (LeNet)	✗	99.11±0.02	**99.17**±0.02	99.13±0.02	**99.16**±0.03	99.19±0.00	99.19±0.00	98.71±0.02	**98.77**±0.03	98.68±0.01	**98.75**±0.00
	✓	99.07±0.02	**99.09**±0.03	98.91±0.02	**99.08**±0.02	99.07±0.06	**99.14**±0.01	98.51±0.05	98.51±0.05	98.54±0.07	**98.74**±0.01
FMNIST (LeNet)	✗	89.35±0.02	**89.65**±0.03	89.92±0.12	**90.12** ±0.03	90.27±0.10	**90.49**±0.12	88.66±0.22	**88.86**±0.18	88.85±0.18	**89.26**±0.21
	✓	**87.43**±0.05	87.37±0.07	88.25±0.14	**88.55**±0.12	88.36±0.13	**88.97**±0.16	87.26±0.07	**87.5**±0.02	87.39±0.12	**87.60**±0.11
FEMNIST (LeNet)	✗	99.71±0.01	**99.79**±0.01	99.48±0.01	**99.53**±0.00	99.58±0.03	**99.62**±0.03	99.52±0.01	**99.76**±0.00	99.64±0.02	**99.68** ±0.01
	✓	95.51±0.02	**96.38**±0.01	94.64±0.01	**95.83**±0.01	94.28±0.04	**94.36**±0.03	94.82±0.01	**96.16**±0.00	94.74±0.01	**96.22**±0.01
CIFAR10 (ResNet)	✗	86.47±0.04	**87.38**±0.03	**87.04**±0.06	86.89±0.14	86.57±0.18	**86.85** ±0.11	87.04±0.12	**87.28**±0.16	87.06±0.11	**87.17**±0.14
	✓	83.06±0.31	**83.21**±0.26	83.65±0.41	**84.24**±0.38	**84.26** ±0.21	83.89±0.18	83.63±0.04	**84.77**±0.03	83.64±0.04	**84.28**±0.04
Quantity Skew											
CIFAR10 (ResNet)	✗	84.78±0.44	**86.28**±0.38	83.81±0.55	**84.42**±0.46	80.58±1.63	**81.81**±1.12	85.92±0.16	**86.58**±0.28	85.98±0.15	**86.47**±0.29

with in-distribution evaluations. We observe that GMA outperforms naive algorithms with sub-sampled clients. Table 4.5 records the performance of the proposed gradient masking on FedAVG, FedProx, FedAdam, and FedYogi, when a fraction of clients is sub-sampled for participation in each communication round. The learning rates that yeilded the performances reported in Tables 4.4 and 4.1 are given in the appendix.

4.4. Quantity skew

In a real federated setting the quantity of data available for update at each client varies drastically. Depending on connectivity, processing power, user behaviour, and various other factors, the number of data samples generated at each client can differ from zero or one data point to an extremely large number. To simulate this quantity imbalance, we have used a Dirichlet distribution based quantity skew with $\alpha = 0.5$ as in [**32**] on CIFAR-10. The sampling done by Dirichlet is independent of the labels or the features of the data samples. The test data contains samples from all classes similar to in-distribution evaluations.

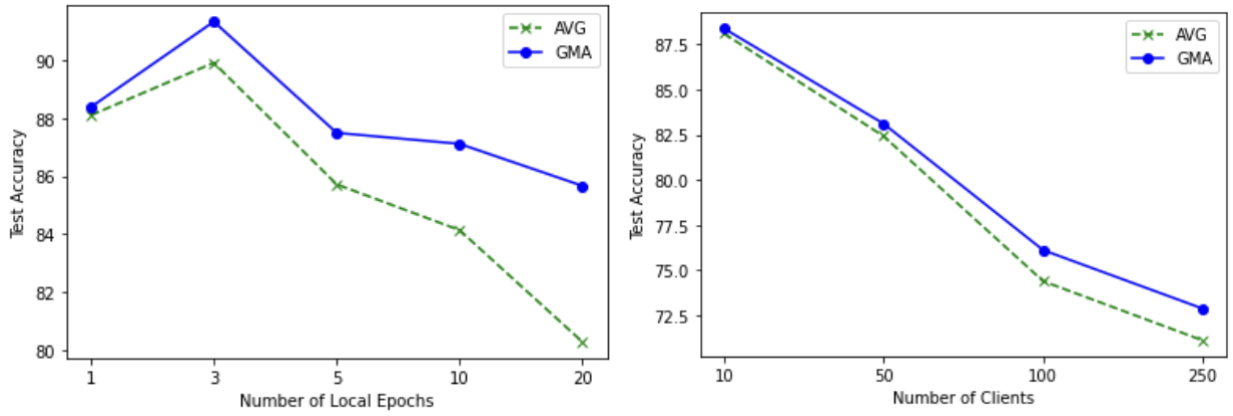
Tableau 4.5. Performance of FedAVG, FedProx, FedAdam, and FedYogi on CIFAR10 and CIFAR100 when C clients are sampled from N for training in each communication round.

Dataset	**Setting**		**FedAVG**		**FedProx**		**FedAdam**		**FedYogi**	
	N	C	AVG	GMA	AVG	GMA	AVG	GMA	AVG	GMA
CIFAR100	500	10	42.68	**44.04**	38.45	**40.74**	47.57	**49.37**	46.27	**48.89**
	100	10	52.65	**52.83**	50.31	**50.76**	53.4	**56.38**	54.64	**56.8**
CIFAR10	500	10	73.38	**74.44**	73.21	**73.95**	76.41	**77.12**	76.13	**77.56**
	100	10	85.55	**85.62**	84.91	**85.37**	85.43	**85.52**	85.62	**85.95**

Tableau 4.6. Performance of FedAVG across a range of global learning rate and client rate on non-iid FMNIST. It can be observed that GMA outperforms AVG in most of the cases where the algorithms learn and converge.

Global Learning Rate							
	0.01	0.1	56.06	61.66	68.02	0.1	AVG
		0.1	**57.72**	**65.13**	**72.56**	0.1	GMA
	0.1	56.93	73.66	83.39	85.22	0.1	AVG
		57.51	**73.9**	**83.79**	**87.22**	0.1	GMA
	1.0	72.69	86.49	87.76	88.31	0.1	AVG
		73.34	**87.0**	**88.14**	**88.4**	0.1	GMA
	1.5	**77.11**	86.29	87.82	86.96	0.1	AVG
		75.63	**87.04**	**88.21**	**88.3**	0.1	GMA
	2.0	71.53	82.42	86.93	87.63	0.1	AVG
		77.59	**86.9**	**87.6**	**88.1**	0.1	GMA
		0.001	**0.01**	**0.05**	**0.1**	**1.0**	

Client Learning Rate



(a) **(b)**

Fig. 4.2. (a) Test accuracy vs. Number of selected clients in the federated network. (b) Test accuracy vs. number of local epochs per client in each communication round. The experiment was on non-i.i.d distributed FMNIST using a LeNet model.

4.5. Convex Objective

To further understand the performance of gradient masking in the convex setting, we experiment with MNIST and FedAVG on both i.i.d and non-i.i.d data distributions and it was observed that GMA outperforms naive averaging in the non-iid setting. A logistic regression model with SGD with momentum optimizer was used at the clients for these experiments. When data distribution was i.i.d, GMA was converging to an average (over last 10 communication rounds) of 92.5% test accuracy while naive averaging obtains to 92.4%. Furthermore, when the data distribution across clients was non-i.i.d, the enhancement in performance was more significant with gradient masking. While naive averaging was converging to 87.0% test and 92.0% train, while GMA reached 88.5% test and 92.2% train. Further demonstrating GMA can generalized better in the non-iid case.

4.6. Client Momentum and Group Norm

In this section we explore the effects of momentum on the client updates and using group norm instead of batch norm on the algorithm performance. Client momentum induces a momentum upon the client updates. This incorporates accumulation of gradients from past steps to determine the direction of updates. Group Normalization is an alternative to Batch

Normalization which outperforms the latter in various computer vision tasks.

4.6.1. Effect of Client Momentum

In contrast to the experiments in [**45**], we use an SGD optimizer with momentum ($\rho = 0.9$) at each client for all our experiments. This was primarily because of the increase in test accuracy observed during our experiments on FedAVG with and without momentum. However, the enhancements due to gradient masking was independent of the momentum induced at the client optimizer. In both cases (with and without momentum), gradient masking was outperforming naive averaging in most of the algorithms and datasets. Table 4.7 shows the performance of the algorithms and their GMA versions on non-i.i.d distributed FMNIST using an LeNet model. The client optimizers used in our experiments is a naive SGD optimizer with momentum parameter and it does not involve the correction parameter introduced in [**64**].

Tableau 4.7. Performance of the algorithms and their GMA versions with and without momentum(ρ) on non-i.i.d distributed FMNIST using an LeNet model. Momentum improves performance of the algorithms. Irrespective of momentum, GMA outperforms AVG.

Dataset (Model)		**FedAVG ($\rho = 0$)**		**FedAVG ($\rho = 0.9$)**	
		AVG	**GMA**	**AVG**	**GMA**
MNIST	**IID**	**99.01**	98.96	99.1	**99.16**
(LeNet)	**Non-IID**	98.43	**98.55**	98.87	**98.9**
FMNIST	**IID**	**88.61**	88.49	89.14	**90.52**
(LeNet)	**Non-IID**	86.95	**87.8**	88.1	**88.38**
FEMNIST	**IID**	98.8	**98.92**	**99.7**	99.68
(LeNet)	**Non-IID**	92.17	**94.61**	94.2	**96.04**
CIFAR-10	**IID**	85.8	**86.31**	87.3	**87.61**
(ResNet)	**Non-IID**	81.1	**82.28**	83.25	**83.95**

Tableau 4.8. Average in-distribution test performance(%) over the last 10 communication rounds of FedAVG, FedProx, SCAFFOLD, FedAdam, FedYogi and their GMA versions on i.i.d and non-i.i.d distributions of CIFAR-10 on ResNet18 models using batch normalization and group normalization. The best result among AVG and GMA versions of each algorithm is shown in bold.

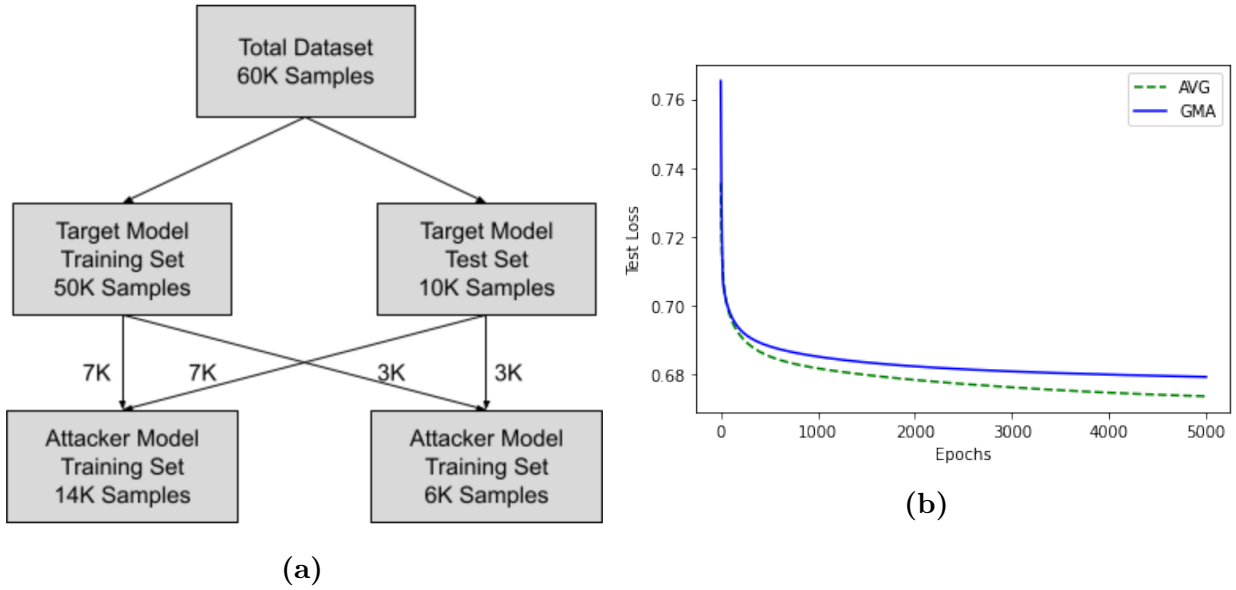
Dataset (Model)		FedAVG		FedProx		SCAFFOLD		FedADAM		FedYogi	
		AVG	GMA	AVG	GMA	AVG	GMA	AVG	GMA	AVG	GMA
ResNet BatchNorm	**IID**	87.11	**87.42**	87.2	**87.38**	87.56	**87.52**	77.32	**80.65**	78.78	**80.55**
	Non-IID	77.3	**79.9**	78.4	**80.2**	75.82	**78.83**	69.82	**74.05**	67.29	**71.51**
ResNet GroupNorm	**IID**	87.3	**87.61**	87.18	**87.5**	86.58	**86.72**	86.9	**87.7**	87.53	**87.78**
	Non-IID	83.25	**83.66**	83.87	**84.4**	84.01	**85.36**	83.53	**84.84**	83.17	**84.55**

4.6.2. Effect of GroupNorm

The test accuracies reported in paper corresponding to CIFAR-10 used a ResNet18 model with batch normalization layers replaced by group normalization[**62**] similar to the experiments in [**45**]. Our initial experiments involved batch normalization as in the original ResNet and it was observed that the replacement of batch norm with group norm improved the test accuracies. Table 4.8 shows the comparison of the algorithms and their GMA versions on CIFAR-10 using ResNet model having batch normalization and group normalization layers. It is to be noted that irrespective of the normalization layer used, gradient masking was outperforming naive averaging across all algorithms and data distributions. This further validates the capabilities of the proposed GMA.

4.7. Membership Inference Attack

Most machine learning models tends to overfit on their training data and such models are susceptible to membership inference attacks that can accurately predict whether a data sample was present in the training set of the model given the model output logits [**55**]. This is a major privacy breach and it can simulated by using a black-box adversarial attacker model. The attacker model we have employed is a binary logistic regression model with binary cross-entropy loss. The input to this attacker model is the logits of the converged gradient masked averaging model and naive averaging model. The attacker model is supposed



(a) **(b)**

Fig. 4.3. (a) Data split and creation for attacker model (b) Test loss vs. epochs of the logistic regression attacker model.

to identify whether the input of the global model corresponding to the logit given was present in the global model's training set or not. For our experiments, we used CIFAR-10 and ResNet models. Firstly, the GMA and AVG models were trained and tested. For each model, the logits corresponding to the train and test set data and their labels (whether train data or test data) were stored. The data is split as shown in Figure 4.3a [**55**] and the attacker model is trained for 5000 rounds. The accuracy is as reported in the paper and loss as shown in Figure 4.3b. A lower attack accuracy of gradient masked implies that GMA has better immunity to membership inference attacks than naive averaging global models.

This experiment is based on [**55**] which suggests that algorithms focusing on learning the causal mechanisms provide stronger privacy guarantees in certain cases, for example they can be more robust to membership inference attacks and model inversion attacks. Based on our experiments we observe that the attack accuracy with respect to the GMA model is 55% while that of naive averaging model is 57%. This suggests that gradient masking can potentially enhance robustness to membership inference attacks

4.8. Conclusion and Future Work

We proposed a new aggregation scheme applicable to a wide variety of federated learning algorithms. The proposed method, gradient masking enhances generalization performance of the global model in FL by focusing on learning the invariances across clients. The simple masking outperforms their naive averaging versions across a variety of algorithms and datasets. Our theoretical analysis shows the convergence of the proposed masking algorithm and the stability of the proposed mask. Future directions include exploration of masks incorporating magnitude and other methods to better capture the invariances, thus leading to better generalization at the global model.

References

[1] Naman Agarwal, Ananda Theertha Suresh, Felix Yu, Sanjiv Kumar et H. Brendan Mcmahan : cpsgd: Communication-efficient and differentially-private distributed sgd, 2018.

[2] Kartik Ahuja, Karthikeyan Shanmugam, Kush R. Varshney et Amit Dhurandhar : Invariant risk minimization games, 2020.

[3] Mohammed Aledhari, Rehma Razzak, Reza M. Parizi et Fahad Saeed : Federated learning: A survey on enabling technologies, protocols, and applications. *IEEE Access*, 8:140699–140725, 2020.

[4] Martin Arjovsky : Out of distribution generalization in machine learning, 2021.

[5] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani et David Lopez-Paz : Invariant risk minimization, 2020.

[6] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal et Karn Seth : Practical secure aggregation for federated learning on user-held data, 2016.

[7] Léon Bottou : Large-scale machine learning with stochastic gradient descent. *In Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[8] Theodora S. Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch. Paschalidis et Wei Shi : Federated learning of predictive models from federated electronic health records. *International Journal of Medical Informatics*, 112:59–67, 2018.

[9] Peter Bühlmann : Invariance, causality and robustness, 2018.

[10] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H. Brendan McMahan, Virginia Smith et Ameet Talwalkar : Leaf: A benchmark for federated settings, 2019.

[11] Yun Hin Chan et Edith C.H. Ngai : FedHe: Heterogeneous models and communication-efficient federated learning. *In 2021 17th International Conference on Mobility, Sensing and Networking (MSN)*. IEEE, dec 2021.

[12] Mingqing Chen, Rajiv Mathews, Tom Ouyang et Françoise Beaufays : Federated learning of out-of-vocabulary words, 2019.

[13] Gregory Cohen, Saeed Afshar, Jonathan Tapson et André van Schaik : Emnist: an extension of mnist to handwritten letters, 2017.

[14] D. Deutsch. : The beginning of infinity: Explanations that transform the world., 2011.

[15] Alireza Fallah, Aryan Mokhtari et Asuman Ozdaglar : Personalized federated learning: A meta-learning approach, 2020.

[16] Sreya Francis, Irene Tenison et Irina Rish : Towards causal federated learning for enhanced robustness and privacy, 2021.

[17] Sreya Francis, Irene Tenison et Irina Rish : Towards causal federated learning for enhanced robustness and privacy. 2021.

[18] Mohamed Gharibi, Srini Bhagavan et Praveen Rao : Federatedtree: A secure serverless algorithm for federated learning to reduce data leakage. *In 2021 IEEE International Conference on Big Data (Big Data)*, pages 4078–4083, 2021.

[19] Anousheh Gholami, Nariman Torkzaban et John S. Baras : Trusted decentralized federated learning. *In 2022 IEEE 19th Annual Consumer Communications Networking Conference (CCNC)*, pages 1–6, 2022.

[20] Sharut Gupta, Kartik Ahuja, Mohammad Havaei, Niladri Chatterjee et Yoshua Bengio : Fl games: A federated learning framework for distribution shifts, 2022.

[21] Jenny Hamer, Mehryar Mohri et Ananda Theertha Suresh : FedBoost: A communication-efficient algorithm for federated learning. *In* Hal Daumé III et Aarti Singh, éditeurs : *Proceedings of the 37th International Conference on Machine Learning*, volume 119 de *Proceedings of Machine Learning Research*, pages 3973–3983. PMLR, 13–18 Jul 2020.

[22] Sungwon Han, Sungwon Park, Fangzhao Wu, Sundong Kim, Chuhan Wu, Xing Xie et Meeyoung Cha : Fedx: Unsupervised federated learning with cross knowledge distillation, 2022.

[23] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith et Brian Thorne : Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption, 2017.

[24] Tzu-Ming Harry Hsu, Hang Qi et Matthew Brown : Measuring the effects of non-identical data distribution for federated visual classification, 2019.

[25] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu et Sen Zhao : Advances and open problems in federated learning, 2021.

[26] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich et Ananda Theertha Suresh : Scaffold: Stochastic controlled averaging for federated learning, 2021.

[27] Masanori Koyama et Shoichiro Yamaguchi : When is invariance useful in an out-of-distribution generalization problem ?, 2021.

[28] A. Krizhevsky et G. Hinton : Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

[29] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol et Aaron Courville : Out-of-distribution generalization via risk extrapolation (rex), 2020.

[30] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol et Aaron Courville : Out-of-distribution generalization via risk extrapolation (rex), 2021.

[31] Y. Lecun, L. Bottou, Y. Bengio et P. Haffner : Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[32] Qinbin Li, Yiqun Diao, Quan Chen et Bingsheng He : Federated learning on non-iid data silos: An experimental study. *arXiv preprint arXiv:2102.02079*, 2021.

[33] Tian Li, Shengyuan Hu, Ahmad Beirami et Virginia Smith : Ditto: Fair and robust federated learning through personalization, 2021.

[34] Tian Li, Anit Kumar Sahu, Ameet Talwalkar et Virginia Smith : Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, May 2020.

[35] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar et Virginia Smith : Federated optimization in heterogeneous networks, 2020.

[36] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson et Blaise Aguera y Arcas : Communication-efficient learning of deep networks from decentralized data. *In Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[37] Mehryar Mohri, Gary Sivek et Ananda Theertha Suresh : Agnostic federated learning, 2019.

[38] Vaikkunth Mugunthan, Ravi Rahman et Lalana Kagal : Blockflow: An accountable and privacy-preserving solution for federated learning, 2020.

[39] Jaehoon Oh, Sangmook Kim et Se-Young Yun : Fedbabu: Towards enhanced representation for federated image classification, 2021.

[40] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele et Bernhard Schölkopf : Learning explanations that are hard to vary, 2020.

[41] Titouan Parcollet, Xinchi Qiu, Daniel J. Beutel, Taner Topal, Akhil Mathur et Nicholas D. Lane : Can federated learning save the planet?, 2021.

[42] Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup et Guillaume Lajoie : Gradient starvation: A learning proclivity in neural networks. *In* A. Beygelzimer, Y. Dauphin, P. Liang et J. Wortman Vaughan, éditeurs : *Advances in Neural Information Processing Systems*, 2021.

[43] Xinchi Qiu, Titouan Parcollet, Daniel J. Beutel, Taner Topal, Akhil Mathur et Nicholas D. Lane : Can federated learning save the planet?, 2020.

[44] Xinchi Qiu, Titouan Parcollet, Javier Fernandez-Marques, Pedro Porto Buarque de Gusmao, Daniel J. Beutel, Taner Topal, Akhil Mathur et Nicholas D. Lane : A first look into the carbon footprint of federated learning, 2021.

[45] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar et H. Brendan McMahan : Adaptive federated optimization, 2021.

[46] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie et Ramtin Pedarsani : Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization, 2020.

[47] Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez et Raman Arora : Fetchsgd: Communication-efficient federated learning with sketching, 2020.

[48] Sumudu Samarakoon, Mehdi Bennis, Walid Saad et Merouane Debbah : Federated learning for ultra-reliable low-latency v2v communications, 2018.

[49] Soroosh Shahtalebi, Jean-Christophe Gagnon-Audet, Touraj Laleh, Mojtaba Faramarzi, Kartik Ahuja et Irina Rish : Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization, 2021.

[50] Sebastian U. Stich : Local SGD converges fast and communicates little. *In International Conference on Learning Representations*, 2019.

[51] Irene Tenison, Sreya Francis et Irina Rish : Gradient masked federated optimization. 2021.

[52] Irene Tenison, Sai Aravind Sreeramadas, Vaikkunth Mugunthan, Edouard Oyallon, Eugene Belilovsky et Irina Rish : Gradient masked averaging for federated learning, 2022.

[53] Irene Tenison, Sai Aravind Sreeramadas, Vaikkunth Mugunthan, Edouard Oyallon, Eugene Belilovsky et Irina Rish : Gradient masked averaging for federated learning, 2022.

[54] Chandra Thapa, M. A. P. Chamikara, Seyit Camtepe et Lichao Sun : Splitfed: When federated learning meets split learning, 2020.

[55] Shruti Tople, Amit Sharma et Aditya Nori : Alleviating privacy attacks via causal learning, 2020.

[56] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos et Yasaman Khazaeni : Federated learning with matched averaging, 2020.

[57] Jianyu Wang et Gauri Joshi : Cooperative sgd: A unified framework for the design and analysis of communication-efficient sgd algorithms, 2019.

[58] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi et H. Vincent Poor : Tackling the objective inconsistency problem in heterogeneous federated optimization, 2020.

[59] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K. Leung, Christian Makaya, Ting He et Kevin Chan : Adaptive federated learning in resource constrained edge computing systems, 2019.

[60] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farokhi Farhad, Shi Jin, Tony Q. S. Quek et H. Vincent Poor : Federated learning with differential privacy: Algorithms and performance analysis, 2019.

[61] Kang Wei, Jun Li, Chuan Ma, Ming Ding, Sha Wei, Fan Wu, Guihai Chen et Thilina Ranbaduge : Vertical federated learning: Challenges, methodologies and experiments, 2022.

[62] Yuxin Wu et Kaiming He : Group normalization, 2018.

[63] Han Xiao, Kashif Rasul et Roland Vollgraf : Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

[64] Jing Xu, Sen Wang, Liwei Wang et Andrew Chi-Chih Yao : Fedcm: Federated learning with client-level momentum, 2021.

[65] Dezhong Yao, Wanning Pan, Yutong Dai, Yao Wan, Xiaofeng Ding, Hai Jin, Zheng Xu et Lichao Sun : Local-global knowledge distillation in heterogeneous federated learning with non-iid data, 2021.

[66] Hao Yu, Sen Yang et Shenghuo Zhu : Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning, 2018.

[67] Honglin Yuan et Tengyu Ma : Federated accelerated stochastic gradient descent, 2021.

[68] Honglin Yuan, Warren Richard Morningstar, Lin Ning et Karan Singhal : What do we mean by generalization in federated learning? *In International Conference on Learning Representations*, 2022.

[69] Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang et Yasaman Khazaeni : Probabilistic federated neural matching, 2019.

[70] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin et Vikas Chandra : Federated learning with non-iid data, 2018.

Appendix A

Learning Rates

Tableau A.1. The best learning rates corresponding to the performances of the algorithms and datasets as reported in Table 4.1

Dataset		FedAVG		FedProx		SCAFFOLD		FedAdam		FedYogi	
		η_g	η_l	η_g	η_l	η_g	η_l	η_g	η_l	η_g	η_l
FEMNIST	**Real World**	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
FedCMNIST	**IID**	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
	Non-IID	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
FedRotMNIST	**IID**	1.0	0.01	1.0	0.01	1.5	0.01	0.05	0.001	0.05	0.001
	Non-IID	1.0	0.01	1.5	0.01	1.0	0.01	0.05	0.001	0.05	0.001

Tableau A.2. The best learning rates corresponding to the performances of the algorithms and datasets as reported in Table 4.4

Dataset		FedAVG		FedProx		SCAFFOLD		FedAdam		FedYogi	
		η_g	η_l	η_g	η_l	η_g	η_l	η_g	η_l	η_g	η_l
MNIST	**IID**	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
	Non-IID	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
FMNIST	**IID**	1.0	0.1	1.0	0.1	1.0	0.01	0.05	0.001	0.05	0.001
	Non-IID	1.0	0.01	1.0	0.01	1.0	0.01	0.05	0.001	0.05	0.001
FEMNIST	**IID**	1.0	0.01	1.0	0.01	1.5	0.1	0.05	0.001	0.05	0.001
	Non-IID	1.0	0.01	1.5	0.01	1.0	0.01	0.05	0.001	0.05	0.001
CIFAR-10	**IID**	1.0	0.01	1.0	0.01	1.5	0.01	0.01	0.001	0.01	0.001
	Non-IID	1.5	0.01	1.0	0.001	1.5	0.001	0.05	0.001	0.05	0.001