Research Scholar: Anuraag Sarkar
Graduate Student Mentor: Yuyang Qiu
Faculty Mentor: Farzad Yousefian

Numerical Validation of Randomized Zeroth-Order Methods for Nonsmooth Federated Learning

RUTGERS
Aresty Research Center for Undergraduates

Abstract

The increasing prevalence of machine learning (ML) in modern applications has motivated the need for distributed learning and data privacy. Additionally, as data sets become larger and more scrambled, it is necessary to develop ML algorithms that can address **heterogeneous** datasets. **Federated learning** (FL) is an emerging ML framework that can address these issues. Rather than having data be sent to a centralized **server** for learning, the individual clients send updated local models to the server. Additionally, FL can achieve adequate accuracies with heterogeneous datasets with differing amounts of and types of datasets. This research aims to numerically validate the capabilities of a recently developed FL method, called **FedRZO**. We validate the algorithm on the MNIST dataset and use ReLU Neural Networks to address the **nondifferentiability** and **nonconvexity** of the **objective function**. Additionally, a **federated MNIST dataset** is used to test the algorithm's effectiveness on both binary classification and multiclass classification. We implement our FL algorithm to test if we obtain **convergence** for the types of classifications. We also provide some preliminary comparisons with other standard ML methods to further test FedRZO's capabilities.

Objectives and FL Background

This research aims to validate the algorithm's effectiveness by:

1. Validating the convergence of FedRZO in training of ReLU neural networks with heterogeneous local data sets for binary classification
2. Extending (1) to multiclass classification
3. Comparing the performance of FedRZO in (1) and (2) with the standard zeroth-order SGD

Background

Federated Learning can be categorized in the following steps:

1. **Initial Step**: Server sends an initial model to all clients
2. **Local Step:** clients individually perform SGD for a certain number of rounds
3. **Aggregation Step:** All clients send their models to the server. The server averages all the models and sends the averaged model to all clients.
4. **Repeat**. The steps repeat until the final step, where a final averaged model is outputted.

Terminology

Server: One entity that averages models
Clients: Individual users that provide data and perform individual local machine learning

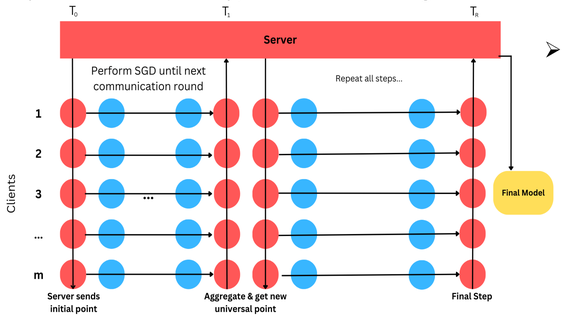
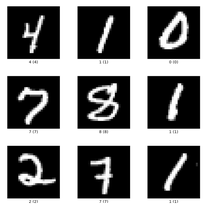


Figure 1: Federated Learning Framework

Dataset and ML Background

MNIST Dataset

In this project, we implement the FedRZO algorithm using the *MNIST* dataset. MNIST is a dataset that contains 70000 images of digits from 0 to 9. Below are examples:



28x28 pixel images

Image from Tensorflow *MNIST* catalog

To validate the capabilities of FedRZO, the framework was used to test its applications in *binary* and *multiclass* classification.

- **Binary** classification is a classification type with two outputs. For MNIST, binary classification can be used to predict if a digit is a certain number or not that number. For instance, a binary classifier on the digit 5 can be used to predict if an image is a 5 or is not a 5.
- **Multiclassification** on MNIST can be used to predict which of the 10 digits an image is. Multiclassification algorithms have 10 possible outputs- one for each digit.

FedRZO was run with zeroth order *stochastic gradient descent (SGD)*, a standard ML algorithm

Neural networks

Neural Networks are a ML model that use SGD to perform ML. For MNIST, Neural Networks take an input of all 784 pixels in the image, perform SGD, and output 10 or 2 numbers, depending on if the algorithm is binary or multiclass. Then, the algorithm generates numbers for all possibilities, and the largest number is the prediction.

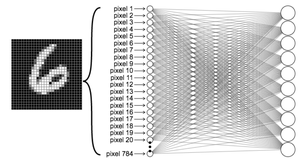


Image from ml4a *Looking Inside Neural Nets*

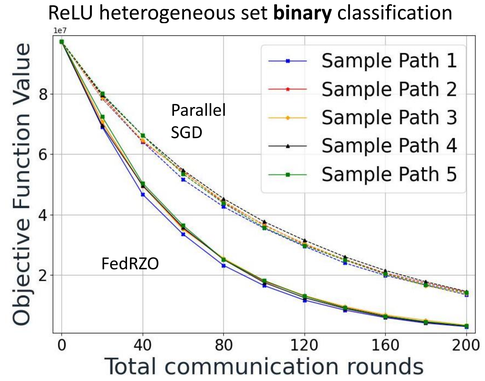
heterogeneous Dataset

A **heterogeneous** dataset is a dataset with differing amounts of images or differing amounts of digit types. This research prepared heterogeneous sets by providing each client with differing amounts of digits.

Specifically, for 5 clients, clients 1 to 5 had 7000, 7000, 14000, 21000, 21000 digits respectively. This ensured that we can validate FedRZO's effectiveness on heterogeneous datasets.

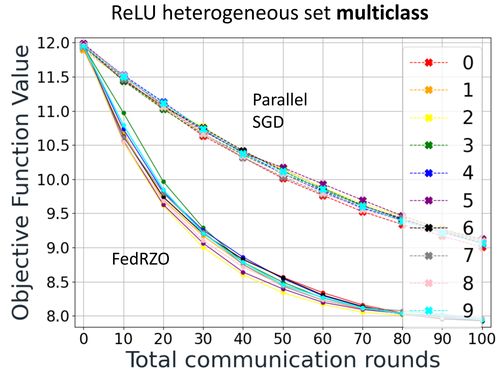
Main Results

SGD (dashed) vs FedRZO (solid) on zeroth order ReLU heterogeneous set **binary** classification



- For binary classification, 5 sample paths were ran to validate consistency of results. The graph depicts convergence for FedRZO faster than the standard SGD ML method for the heterogeneous dataset. Because FedRZO converges faster, for heterogeneous datasets, FedRZO will reach higher accuracies faster than standard ML methods.

SGD (dashed) vs FedRZO (solid) on zeroth order ReLU heterogeneous set **multiclass**



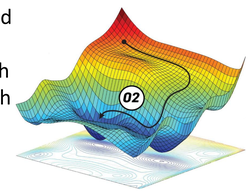
- In the above graph, FedRZO was compared to SGD for One vs Rest Multi Classification on a heterogeneous dataset. The results indicate that FedRZO converged faster than SGD for all ten digits. Therefore, FedRZO will reach higher accuracies faster than standard ML methods.

Discussion and Insights

To test the convergence of FedRZO, the algorithm was compared to stochastic gradient descent (SGD). Comparisons were done by plotting the convergence of both algorithms.

By plotting the objective function value over a period of rounds, we can see how the algorithms converge to a minimum value. Figure 2 (underneath) illustrates a descent algorithm that converges to a local minimum. The faster the convergence, the better the algorithm is for ML. Convergence was tested for both binary and multiclass classification.

The algorithms were implemented on a neural network model with the ReLU activation function. Both algorithms are zeroth order, which uses evaluations of objective function rather than gradients. Convergence was validated on a federated MNIST dataset distributed among 5 clients.



[Figure 2] Image from Paperspace *Gradient Descent with Python* blog series

FedRZO Algorithm

To the right is the FedRZO algorithm designed by the lab **[Reference 3]**. The algorithm will be implemented in python and is validated in this research.

Algorithm 1 Randomized Zeroth-Order Locally-Projected Federated Averaging ($\text{FedRZO}_{\text{nn}}$)

input: Server chooses a random initial point $\hat{x}_0 \in X$, stepsize γ, smoothing parameter η, synchronization indices $T_0 := 0$ and $T_r \geq 1$, where $r \geq 1$ is the communication round index
for $r = 0, 1, \ldots$ **do**
  Server broadcasts $\hat{x}_r$ to all clients: $x_{i,T_r} := \hat{x}_r$, $\forall i \in [m]$
  for $k = T_r, \ldots, T_{r+1} - 1$ **do in parallel by clients**
    Client i generates the random replicates $\xi_{i,k} \in \mathcal{D}_i$ and $v_{i,k} \in \eta\mathbb{S}$
    $g^{\eta}_{i,k} := \frac{n}{\eta^2}\left(\tilde{f}_i(x_{i,k} + v_{i,k}, \xi_{i,k}) - \tilde{f}_i(x_{i,k}, \xi_{i,k})\right) v_{i,k}$
    Client i does a local update as $x_{i,k+1} := x_{i,k} - \gamma\left(g^{\eta}_{i,k} + \frac{1}{\eta}\left(x_{i,k} - \mathcal{P}_{X_i}(x_{i,k})\right)\right)$
  end for
  Server receives $x_{i,T_{r+1}}$ from all clients and aggregates, i.e., $\hat{x}_{r+1} := \frac{1}{m}\sum_{i=1}^{m} x_{i,T_{r+1}}$
end for

Future Direction

- Validate the effectiveness of FedRZO on one-vs-one multiclass classification.
- Use FedRZO for predictive learning on other datasets, such as EMNIST.
- Collect data from differing number of clients and observe how changing the number of clients affects convergence.

References

[1] Géron, Aurélien. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.", 2022.
[2] McKinney, Wes. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython.* " O'Reilly Media, Inc.", 2012.
[3] Qiu, Yuyang, Uday V. Shanbhag, and Farzad Yousefian. "Zeroth-Order Methods for Nondifferentiable, Nonconvex, and Hierarchical Federated Optimization." Ongoing work.
[4] McMahan, Brendan, et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Concluding Remarks

The results indicate that FedRZO is an effective algorithm for ML with heterogeneous datasets. Additionally, FedRZO can effectively learn by only collecting models from each client, rather than user data. Additionally, FedRZO's compatibility with both binary and multiclass classification proves promising for further applications of the algorithm.

Acknowledgements

- We greatly appreciate Aresty Summer Science Program's supportive role in this research.
- We acknowledge the support of the Department of Energy under the grant #DE-SC0023303