Weekly Report (April 1, 2023 - April 9, 2023)

Zixun
XXX
XXX
zixunxiong@umass.edu

Abstract

The focus of this project was on adaptive federated learning algorithms(adaptive FL algorithms). I want to investigate the limitations of adaptive FL algorithms. Thus, I ran some tests on both accuracies on different public datasets and their performance in adversarial cases when facing attacks like label flipping. The experiments show that despite their promising performance in accuracy, adaptive FL algorithms do face some issues when the learning rate is improper or in adversarial cases. In the last part, I describe my future plans according to the findings of this project.

1 Introduction

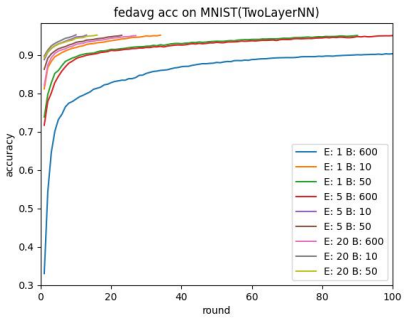
In [2], Reddi. et al. claim that their adaptive federated algorithms achieve great performance compared with non-adaptive ones, Figure 1 in this paper implied unstable performances of adaptive methods like FedADAGRADE. What's more, the whole test is based on a non-adversarial basis. Considering its mechanics it's different from non-adaptive algorithms like FedAVG, highlighting the need for new defenses for non-adaptive federated learning. In the following parts, I ran experiments on different algorithms to make further observations. Due to limitations on computation resources and time, I only ran FedAVG[1], FedOPT, and FedADAGRADE on the MNIST dataset. For security issues, I ran a label-flipping attack by varying different ratios of malicious clients. I have also implemented Krum and data loaders for the Shakespeare dataset, all codes can be found on here.

2 Experiments evaluation: datasets, tasks, and methods

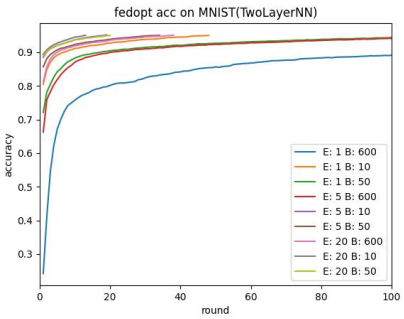
3 Experimental evaluation: results

3.1 Test accuracy experiments

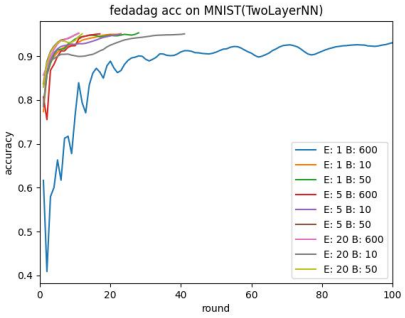
I ran test accuracy experiments on the MNIST dataset to make comparisons between different aggregation algorithms. As indicated in Figure 1, with a higher number of rounds on clients(E) and smaller batch size(B), all aggregation algorithm achieves better accuracy and convergence. Each figure is a test accuracy vs round of communication plot. For both adaptive federated learning algorithms (e.g. FedOPT, FedADAGRADE) and non-adaptive algorithms(e.g. FedAVG), I ran grid-search tests on the validation dataset(for MNIST dataset, it's the test dataset) to fine-tune hyper-parameters like server learning rates and clients learning rates. Thereby, all algorithms are guaranteed to achieve the greatest accuracy to make sure the comparison made is based on a fair basis.



(a) test acc on MNIST(FedAVG)



(b) test acc on MNIST(FedOPT)

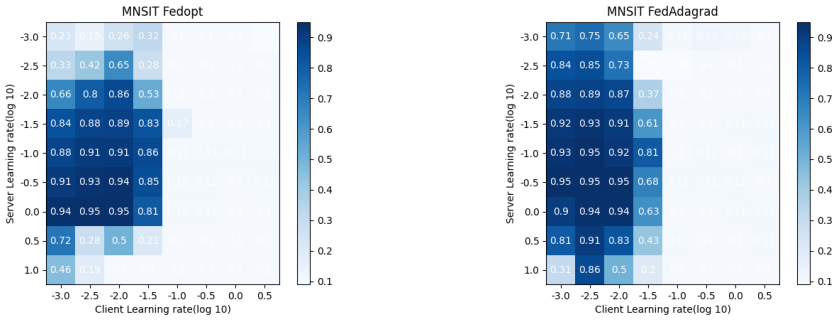


(c) test acc on MNIST(FedADAGRADE)

Figure 1: test accuracy on MNIST dataset within 100 epoch and fine-tuned learning rates and various E and B

It's not hard to find in 1(c) that FedADAGRADE shows a higher accuracy and early stop with proper E and B. What needs to be added is, with improper hyper-parameters, the algorithm is relatively unstable.

3.2 Grid search



(a) fedopt

(b) fedadagrade

Figure 2: grid search on MNIST

To get fine-tuned learning rates, I ran gird searches on the validation set by varying hyper-parameters. As shown in 2, FedADAGRADE it's easier to tune. By selecting learning rates for both client and server datasets with the best test accuracy, we get the fine-tuned FedOPT and FedADAGRADE.

3.3 Impact of compromised ratio

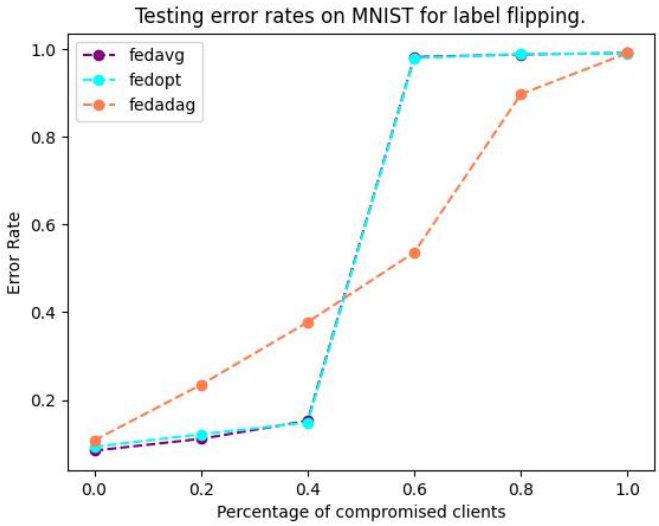


Figure 3: impact of compromised clients ratio(20 epochs)

As implied in 3, when the ratio of comprised clients is relatively low(which is close to the real-world settings), FedADAGRADE has a higher error rate. It might be the case that adaptive learning is more sensitive to outliers.

4 Future Plan

As shown in part 3, FedADAGRADE doesn't have estimated performance when facing malicious scenarios, but it's still not sure whether it's caused by using a small dataset or running inadequate epochs. Thus, I'm going to do more experiments with higher epochs and bigger datasets. For security issues, I plan to try some more attacks and defenses with non-adaptive learning. What's more, I also plan to design attacks targeted at adaptive FL algorithms and defenses accordingly(e.g. changing Krums into defenses like picking deltas with the highest local similarity).

References

[1] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics (pp. 1273-1282). PMLR.

[2] Reddi, S., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., ... & McMahan, H. B. (2020). Adaptive federated optimization. arXiv preprint arXiv:2003.00295.