Efficient Byzantine-Resilient Stochastic Gradient Descent

Kaiyun Li [1], Xiaojun Chen [1], Ye Dong [1], Peng Zhang [2], Dakui Wang [1] and Shuai Zeng [1]

[1] Institute of Information Engineering, Chinese Academy of Sciencess, Beijing, China.
[2] Guangzhou University, Guangzhou, China.

INTRODUCTION

Federated Learning often suffers from Byzantine failures. Some previous works, albeit workable under Byzantine failures, have the shortcomings of either a sub-optimal convergence rate or high computation cost. **To this end, we propose a new Byzantine-resilient stochastic gradient descent algorithm (BrSGD for short) which is provably robust against Byzantine failures.**

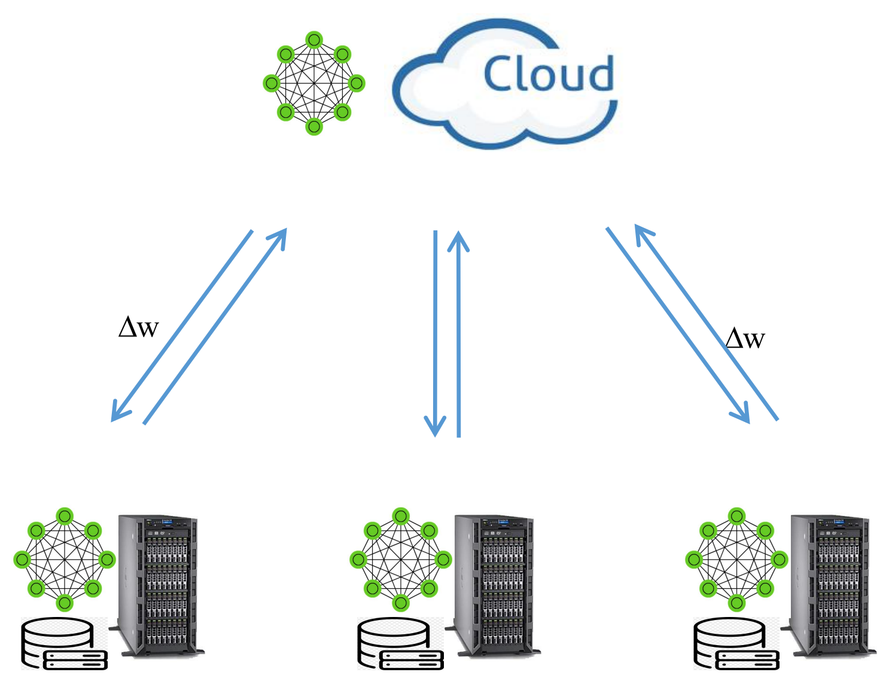
Algorithms	Krum[1]	Median[2]	Trimmed[3]	BrSGD
Computation Complexity	O(m^2d)	O($dm\log m$)	O($dm\log m$)	O(md)

The contributions can be summarized as follows:

- We present a new robust aggregation rule for distributed synchronous Stochastic Gradient Descent algorithm BrSGD in an adversarial setting. The new algorithm can handle Byzantine resilience. Moreover, the computation complexity of the proposed algorithm is O(md).
- We theoretically analyze the statistical error rates of our method on strongly convex loss functions. In particular, our algorithm can achieve an order-optimal statistical error rate for strongly convex losses.
- We also demonstrate the convergence of the proposed algorithm by conducting empirical studies. The experimental results on four types of Byzantine attacks match the results of none-Byzantine machines in terms of effectiveness and convergence.

METHODS

The master machine broadcasts the initialized model, and all the workers receive and accept it as the local initialized model. Then, the master and the workers update the model iteratively：



① At each iteration, the normal workers compute the gradients of their local loss functions and then upload to the master. The Byzantine workers may send any messages.

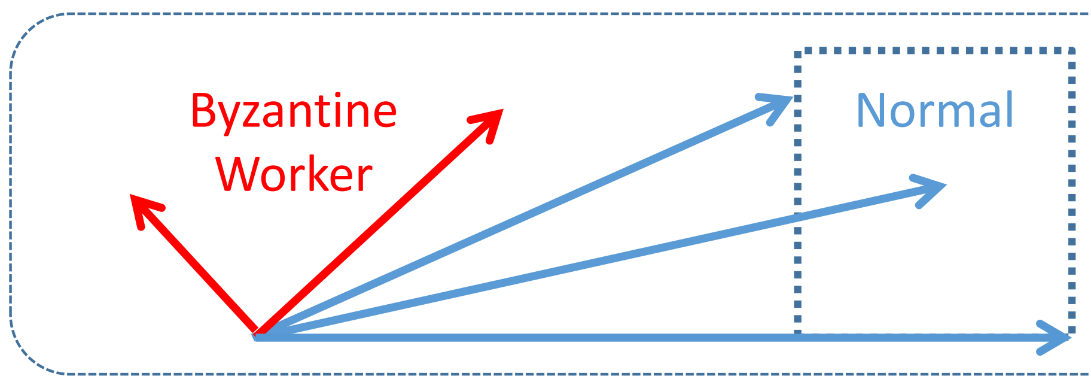
② Next, the master performs a **robust aggregator** to compute gradients for model updating and send the gradient to all workers.

③ After receiving the gradient from the master, the workers update its parameters in the way of gradient descent and moves into the next iteration until the whole algorithm is completed.

Pic 1: The overview of BrSGD archiecture.

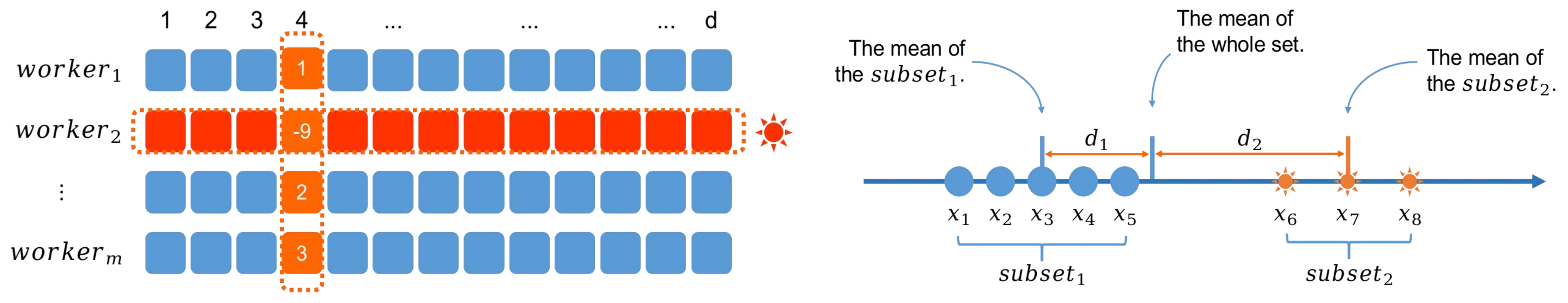
we focus on the **aggregation** method construction. Intuitively, the aggregation rule should return a vector that is not too far from the "real" gradient.

Constraint 1



$$C_1 = \{i : \|g^i - g^{med}\|_1 \le 2\Im\}$$

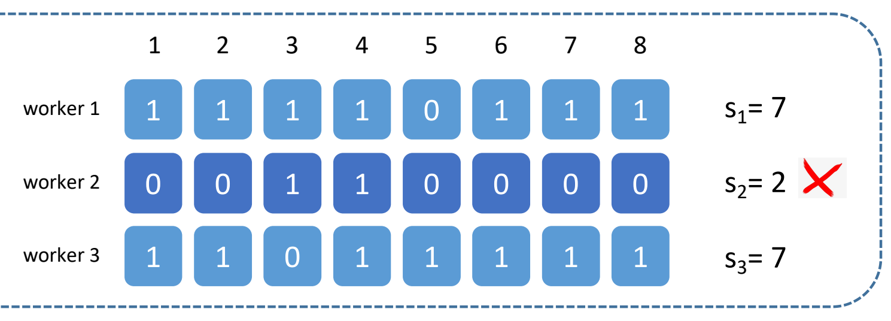
Pic 2: let g^i be a stochastic gradient computed by a worker i, then g^i shall concentrate around g^{med}, where g^{med} is the median value of g^i.


1
2
3
4
...
...
...
d
worker$_1$
worker$_2$
⋮
worker$_m$
1
-9
2
3
The mean of the $subset_1$.
The mean of the whole set.
The mean of the $subset_2$.
d_1
d_2
x_1 x_2 x_3 x_4 x_5
x_6 x_7 x_8
$subset_1$
$subset_2$

Pic 3: The gradients are collected and combined into a matrix G. We divide one column of data into two subsets using their mean and set the subset with a large number of elements to 1, otherwise 0. Finally, a scoring matrix M composed of 0 and 1 will be generated, and the result of adding the rows will be used as a score for the corresponding gradient.

Constraint 2

$$C_2 = \{i : \mathrm{Top}^{\beta}_{\max}(\sum_{j\in[d]} M_{i,j}, i \in [m])\}$$



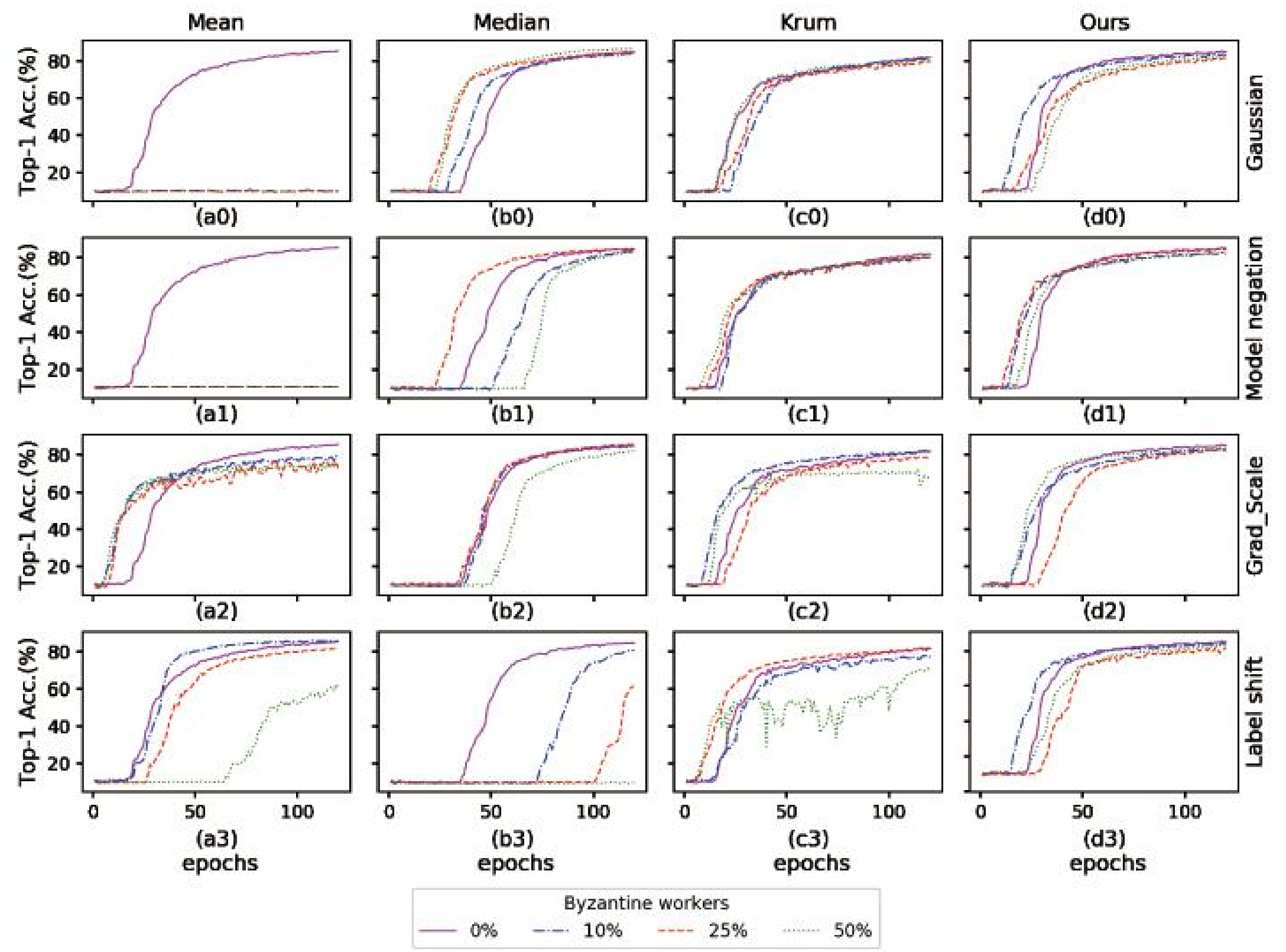
	1	2	3	4	5	6	7	8	
worker 1	1	1	1	1	0	1	1	1	s_1= 7
worker 2	0	0	1	1	0	0	0	0	s_2= 2 ✗
worker 3	1	1	0	1	1	1	1	1	s_3= 7

Finally,

$$g = mean(g^i, i \in C_1 \cap C_2)$$

RESULT

The results w.r.t. accuracy vary with the epochs under Byzantine attacks, where we set step-size η=0.03 for the LeNet on FashionMNIST. Curves correspond to losses, and columns correspond to different aggregation methods. Byzantine workers perform four attacks with 10%, 25%, and 50% attackers, respectively. We set β=1/2 for our algorithm.



In conclusion, the distributed gradient descent algorithm suffers from severe performance loss in adversarial settings. Moreover, show our algorithm can indeed defend against Byzantine failures. Compared with the median-based and *Krum* methods, our algorithm has excellent accuracy and faster convergence for the four attacks.

CONCLUSION

- In this paper, we study a new distributed stochastic optimization algorithm in an adversarial setting with the purpose of obtaining the optimal statistical results and computation efficiency simultaneously.
- Based on the honest-majority assumption, we propose a new stochastic gradient descent algorithm BrSGD. We show that the method can achieve an order-optimal

$$\triangle = \tilde{O}\left(\frac{1}{\sqrt{n}} + \frac{1}{\sqrt{nm}}\right)$$

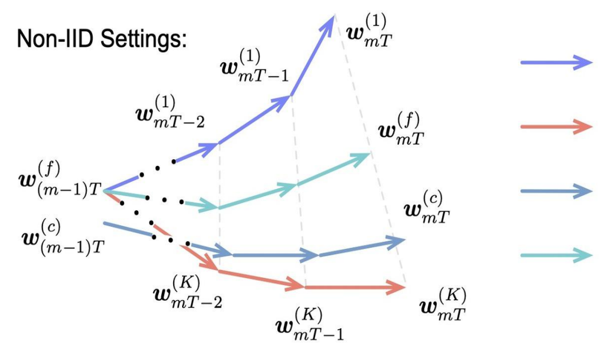
for strongly convex losses, and the computation complexity of our algorithm is O(md).

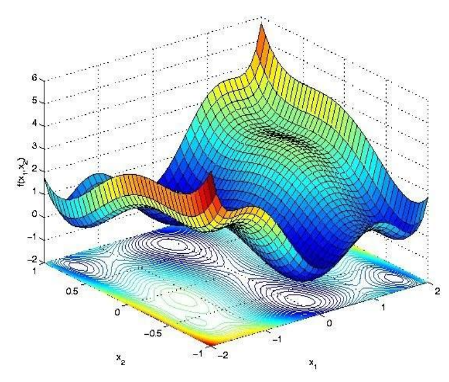
- Moreover, we conduct extensive experiments to show that our method outperforms the state-of-the-art methods in terms of effectiveness and convergence.

FUTURE WORK

1. Non-IID Settings

2. Non-Convex loss

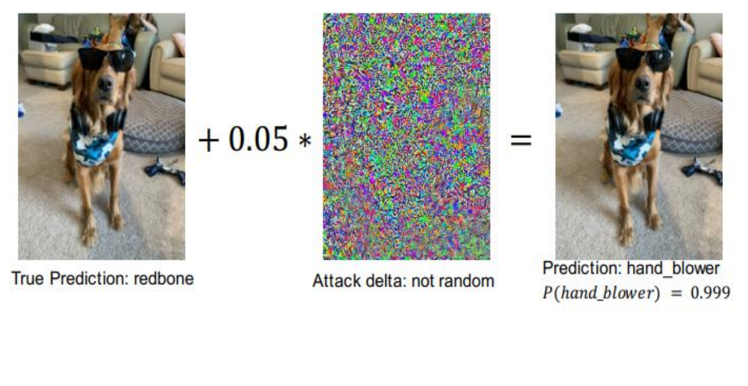




3. Privacy protection

4. Model Security

Secure Multi Party Computation



REFERENCES

[1] Blanchard, Peva, et al. "Machine learning with adversaries: Byzantine tolerant gradient descent." Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017.

[2] Chen, Yudong, Lili Su, and Jiaming Xu. "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent." Proceedings of the ACM on Measurement and Analysis of Computing Systems 1.2 (2017): 1-25.

[3] Yin, Dong, et al. "Byzantine-robust distributed learning: Towards optimal statistical rates." International Conference on Machine Learning. PMLR, 2018.