DISTRIBUTED ONLINE LEARNING WITH ADVERSARIAL PARTICIPANTS IN AN ADVERSARIAL ENVIRONMENT

Paper ID: 1869

ICASSP 2023
4 - 10 JUNE, RHODES ISLAND, GREECE

Xingrong Dong[1] Zhaoxian Wu[1] Qing Ling[1] Zhi Tian[2]
[1]Sun Yat-Sen University [2]George Mason University

Background

- Online Learning
  - Online learning is a powerful tool to process streaming data.
  - In response to an environment that provides (adversarial) losses sequentially, an online learning algorithm makes one-step-ahead decisions.
- Distributed Online Learning
  - Multiple participants separately collect streaming data, make local decisions.
  - Server aggregates all local decisions to a global one.
  - Applications: online web ranking and advertisement recommendation.
- Performance of an online learning algorithm is characterized by (adversarial) regret, and a sublinear (adversarial) regret is perferred.

Adversarial participants

- But adversarial (Byzantine) participants may exist, which can collude and arbitrarily modify the messages sent to server (called the Byzantine Attacks).
- Is it possible to develop a Byzantine-robust distributed online learning algorithm with provable sublinear adversarial regret, in an adversarial environment and in the presence of adversarial participants ?
- Answer is Negative
  - × Distributed online gradient descent with mean: infinite adversarial regret.
  - × Even with robust aggregation rules: linear adversarial regret.

Problem Formulation: Adversarial Regret

- Consider n participants in $\mathcal{N}$, h honest in $\mathcal{H}$, b Byzantine in $\mathcal{B}$, $n = h + b$.
- Suppose the ratio of Byzantine participants is less than half: $\alpha := \frac{b}{n} < \frac{1}{2}$.
- Goal: minimize **adversarial regret** over T steps

$$R_T := \frac{1}{h}\sum_{t=1}^{T}\sum_{j\in\mathcal{H}} f_t^j(w_t) - \min_{w\in\mathbb{R}^d}\frac{1}{h}\sum_{t=1}^{T}\sum_{j\in\mathcal{H}} f_t^j(w), \tag{1}$$

and f_t^j is the loss revealed to $j \in \mathcal{H}$ at the end of step t.

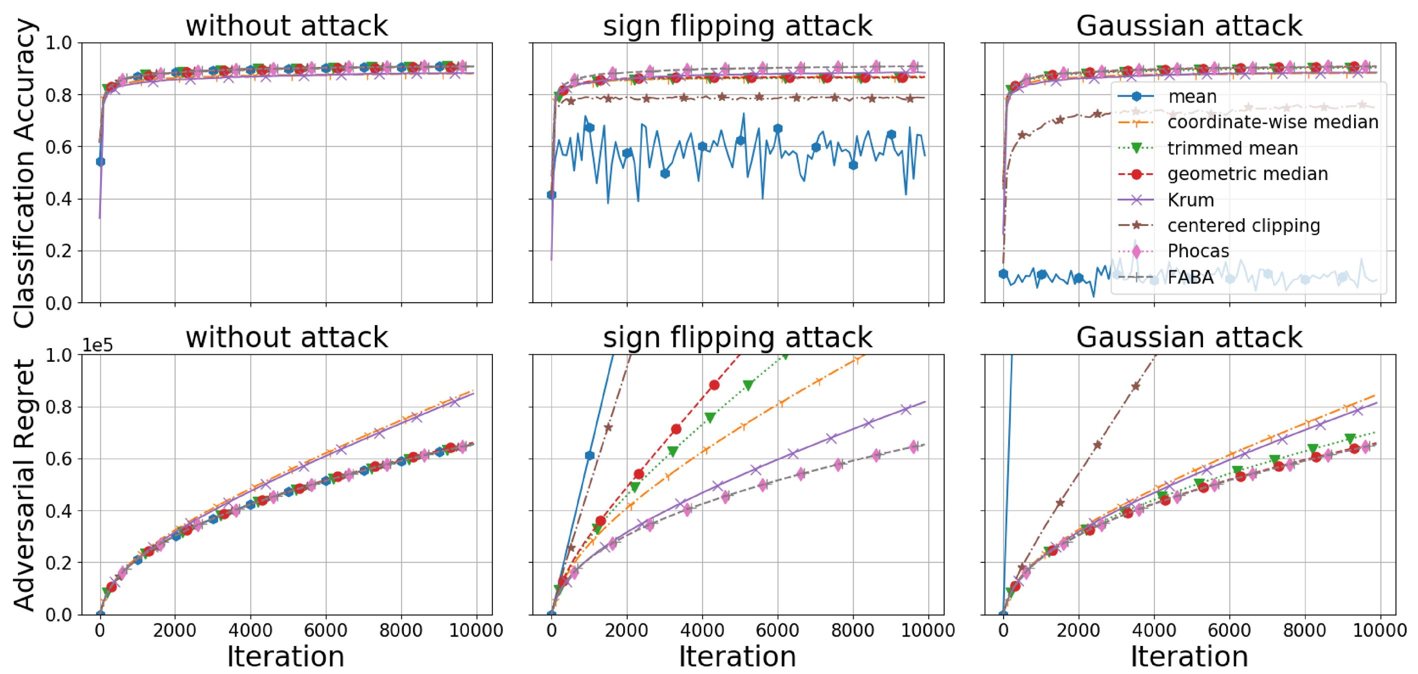

Fig. 1. Performance of Byzantine-robust distributed online gradient descent.

Byzantine-robust Distributed Online Gradient Descent

Adversarial Regret & Algorithm

Each honest participant j makes its local decision by **online gradient descent**:

$$w_{t+1}^j = w_t - \eta_t \nabla f_t^j(w_t), \quad \text{step size } \eta_t > 0. \tag{2}$$

- **Baseline**: distributed **online gradient descent** (2) with mean aggregation
  Server aggregates messages z_t^j (w_t^j from honest and arbitrary from Byzantine)

$$w_{t+1} = \frac{1}{n}\sum_{j=1}^{n} z_{t+1}^j. \tag{3}$$

- **Ours**: Byzantine-robust distributed **online gradient descent** (2) with AGG

$$w_{t+1} = AGG(z_{t+1}^1, z_{t+1}^2, \cdots, z_{t+1}^n). \tag{4}$$

AGG is Robust Bounded Aggregation, if

$$\|w_t - \bar{z}_t\|^2 = \|AGG(z_t^1, z_t^2, \cdots, z_t^n) - \bar{z}_t\|^2 \le C_\alpha^2\zeta^2, \quad \bar{z}_t := \frac{1}{h}\sum_{j\in\mathcal{H}} z_t^j, \tag{5}$$

where $\|\bar{z}_t - z_t^j\|^2 \le \zeta^2$, C_α is a constant depending on α and aggregation rules.

Assumptions

Define $\nabla \bar{f}_t(w_t) := \frac{1}{h}\sum_{j\in\mathcal{H}} \nabla f_t^j(w_t)$ and $w^* := \arg\min_{w\in\mathbb{R}^d}\sum_{t=1}^{T} f_t(w)$. For any honest participant's loss f_t^j where $j \in \mathcal{H}$ and any $x, y \in \mathbb{R}^d$, we assume

1. L-smoothness. $||\nabla f_t^j(x) - \nabla f_t^j(y)|| \le L||x-y||$.
2. μ-strong convexity. $\langle \nabla f_t^j(x), x-y\rangle \ge f_t^j(x) - f_t^j(y) + \frac{\mu}{2}\|x-y\|^2$.
3. Bounded deviation. $||\nabla f_t^j(w_t) - \nabla \bar{f}_t(w_t)||^2 \le \sigma^2$.
4. Bounded gradient at the overall best solution. $\|\frac{1}{h}\sum_{j\in\mathcal{H}}\nabla f_t^j(w^*)\|^2 \le \xi^2$.

Convergence

Theorem 1: Under Assumptions 1, 2, 3 and 4, if $\eta = \mathcal{O}(\frac{1}{\sqrt{T}})$, Byzantine-robust distributed online gradient descent has a linear adversarial regret bound

$$R_T = \mathcal{O}((C_\alpha^2\sigma^2 + \xi^2)\sqrt{T}) + \mathcal{O}(C_\alpha^2\sigma^2 T). \tag{6}$$

We construct a counter-example to demonstrate $\mathcal{O}(\sigma^2 T)$ is tight.

How to derive sublinear regret under Byzantine Attacks?
→ Not fully adversarial environment.

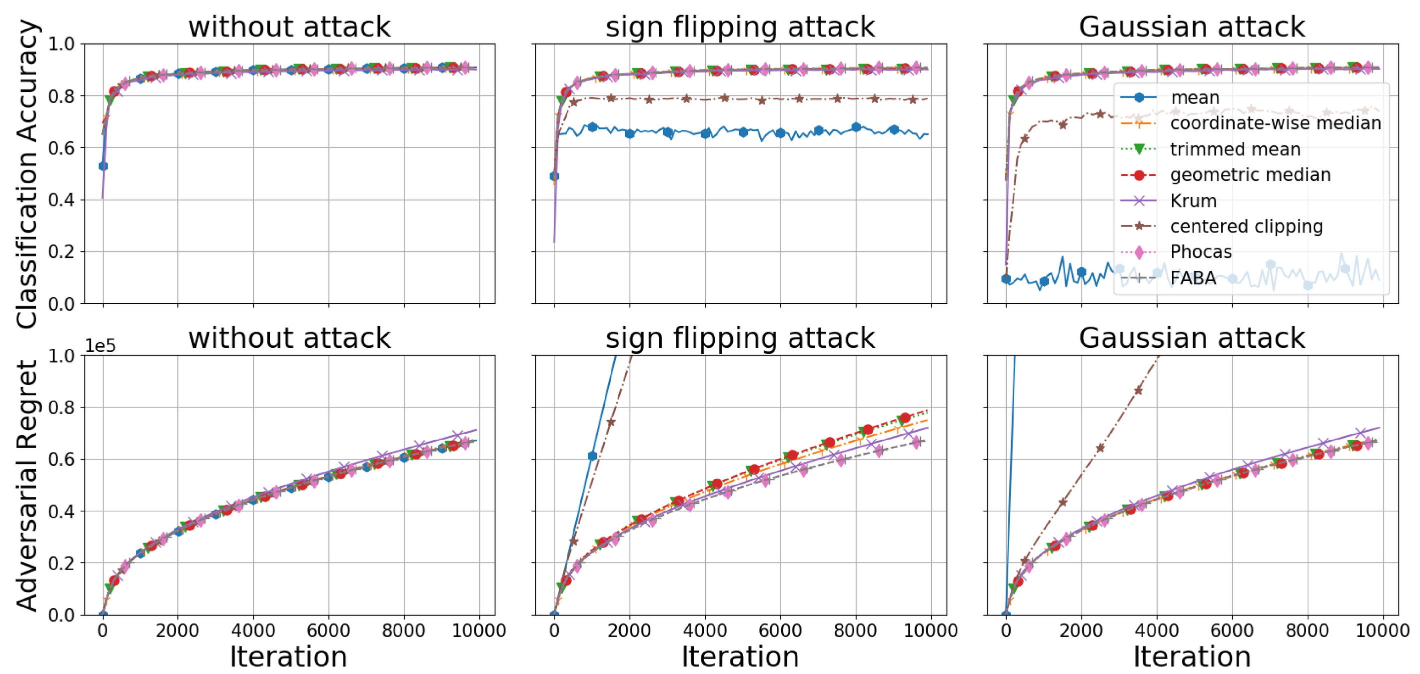

Fig. 2. Performance of Byzantine-robust distributed online momentum.

Byzantine-Robust Distributed Online Momentum

Stochastic Regret & Algorithm

- Not fully adversarial environment: losses are independent and identically distributed (i.i.d.), meaning $f_t^j \sim \mathcal{D}$ for all $j \in \mathcal{H}$ and all t.
- Define the expected loss $F(w) := \mathbb{E}_{\mathcal{D}} f_t^j(w)$ for all $j \in \mathcal{H}$ and all t.
- New Goal: minimize **stochastic regret** over T steps

$$S_T := \mathbb{E}\sum_{t=1}^{T} F(w_t) - T\cdot\min_{w\in\mathbb{R}^d} F(w). \tag{7}$$

- Each honest participant j maintains a momentum vector to reduce variance

$$m_t^j = \nu_t\nabla f_t^j(w_t) + (1-\nu_t)m_{t-1}^j, \tag{8}$$

where $0 < \nu_t < 1$ is momentum parameter. Then, it makes a local decision

$$w_{t+1}^j = w_t - \eta_t m_t^j. \tag{9}$$

- **Ours**: Byzantine-Robust distributed **online momentum** (9) with AGG.

Assumptions

For expected loss $F(w)$ and any $x, y \in \mathbb{R}^d$, we assume

5. L-smoothness. $||\nabla F(x) - \nabla F(y)|| \le L||x-y||$.
6. μ-strong convexity. $\langle \nabla F(x), x-y\rangle \ge F(x) - F(y) + \frac{\mu}{2}\|x-y\|^2$.
7. Bounded variance. $\mathbb{E}_{\mathcal{D}}||\nabla f_t^j(w_t) - \nabla F(w_t)||^2 \le \sigma^2$.

Convergence

Theorem 2: Supposed losses are i.i.d., under Assumptions 5, 6 and 7, if $\eta = \mathcal{O}(\frac{1}{\sqrt{T}})$ and $\nu = \mathcal{O}(\frac{1}{\sqrt{T}})$, Byzantine-robust distributed online momentum has a sublinear stochastic regret bound

$$S_T = \mathcal{O}\left(\left(1 + \frac{\sigma^2}{h}\left(1 + (h+1)C_\alpha^2\right)\frac{L^4}{\mu^4}\right)\sqrt{T}\right). \tag{10}$$

Numerical Experiments

Setting

- Softmax regression on the i.i.d. MNIST dataset.
- Measurement: adversarial regret and accuracy.

Observations from Experiments

- **Fig. 1**: Byzantine-robust distributed online gradient descent shows robustness.
- **Fig. 2**: Byzantine-robust distributed online momentum shows improvement.

More experimental results on non-i.i.d. data are shown in the paper.

More results and codes are available at https://github.com/wanger521/OGD.