Analyzing Federated Learning through an Adversarial Lens

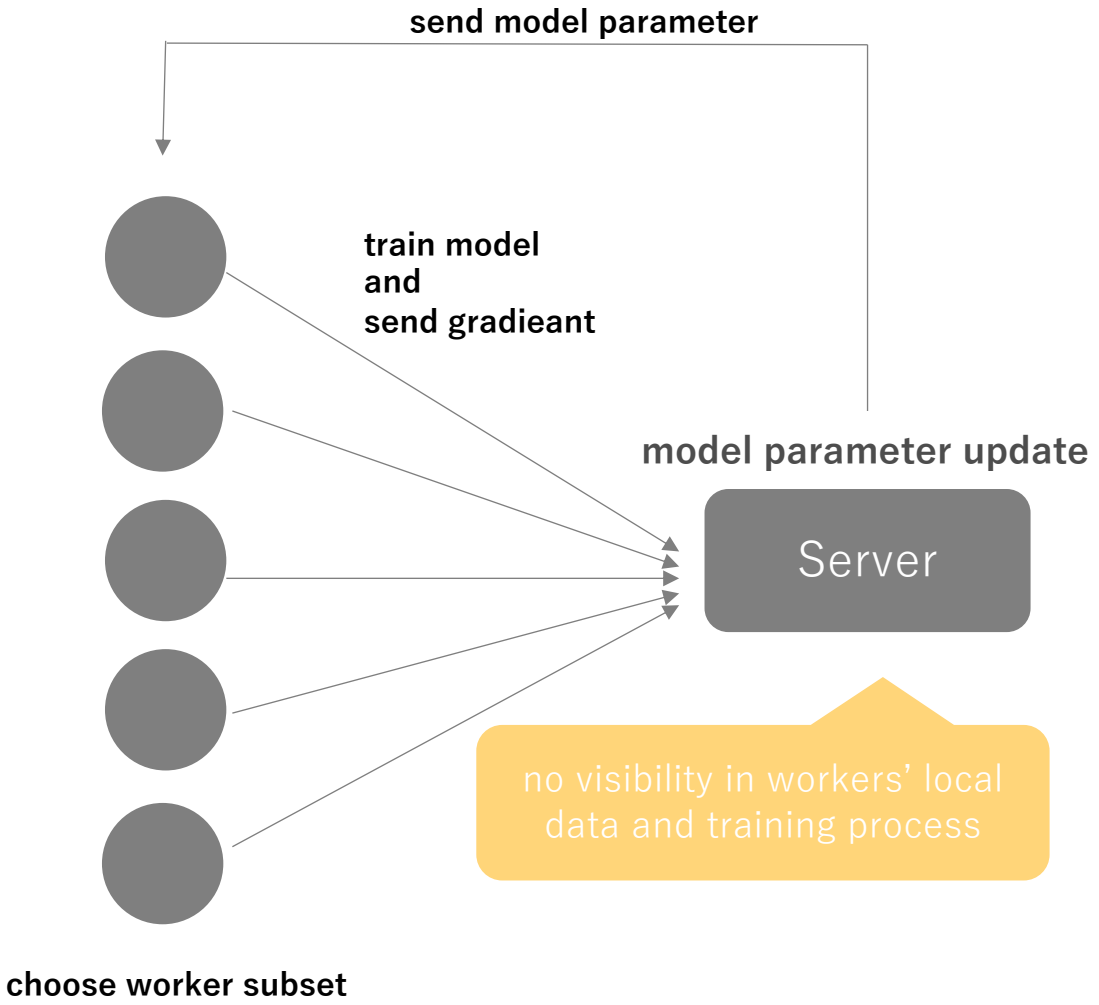
Bhagoji, N. Chakraborty, S. Mittal, P. and Calo, S.

In *Proceedings of the 36th International Conference on Machine Learning*, pages 634–643, 2019. (ICML2019)

Introduction

What is Federated learning ?

- Federated learning [1]とは, 各ワーカーのローカルなデータを使用して学習するが, モデルのパラメータ更新のみを共有する学習手法
- データが元々分散して存在している・データが膨大で分散して学習を行わないと計算資源が不足してしまう状況から生まれた手法



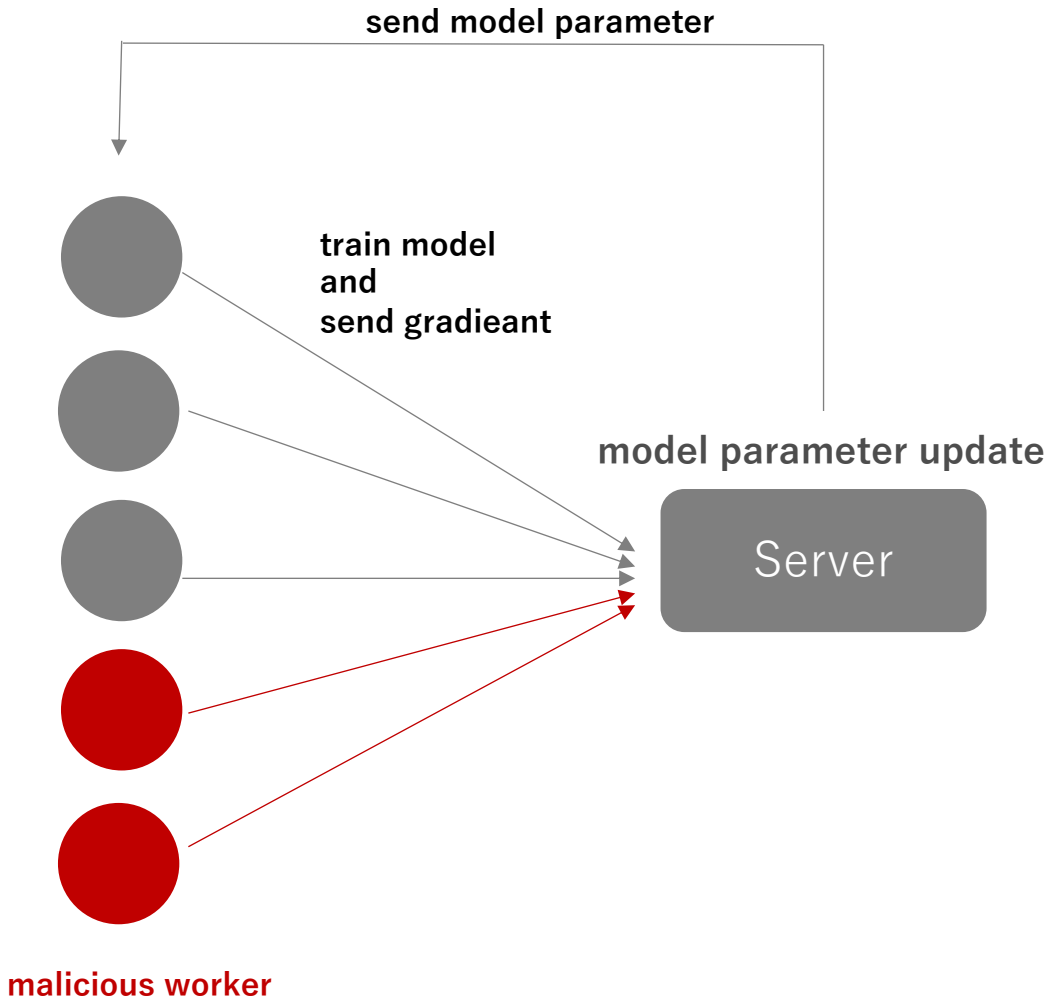
[1] McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 2017.

Introduction

Model Poisoning : new threat for federated learning

- 少量の悪意のあるワーカーによって引き起こされ, 特定の入力に対して高い確信度で誤分類するようにすることが目的
- ステルス性 (正常なワーカーに近い挙動)と敵対的な目的 (特定の入力に対して誤分類)を交互最小化する

- 既存のビザンチン障害に耐えうる集約規則に対して堅牢でないことを経験的に示す
- Federated learningの脆弱性を主張



Model poisoningと関連する攻撃手法との比較

Attack name	Adversarial Examples	Byzantine failure	Model poisoning
method	**入力を敵対的に変更**	学習プロセスを変更	**学習プロセスを変更**
Object	特定の入力に対してパフォーマンスを低下**(学習後)**	**パフォーマンスを低下**	**特定の入力に対してパフォーマンスを低下**

Introduction

Contributions

- Federated learningに対して, targeted model poisoning が可能であるような攻撃手法の設計した.
- 評価実験から単一の悪意あるエージェントを制御することでグローバルモデルに収束しながら, 特定のデータに対してほぼ確実に誤分類を達成することを確認した.
- 既存のビザンチン障害を防ぐ防御手法 (Krum [2]・coordinate-wise median [3])を使用してもtargeted model poisoningが高い確信度で成功してしまうことを実験的に示した.

[2] Blanchard, P., El Mhamdi, E. M., Guerraoui, R., and Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in Neural Information Processing Systems, 2017.
[3] Yin, D., Chen, Y., Ramchandran, K., and Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. arXiv preprint arXiv:1803.01498, 2018

Federated Learning and Model Poisoning

Federated Learning Formulation

時刻 t におけるFederated learningの動作

Worker

1. K人のワーカーからk人のワーカーの部分集合を選択
2. 時刻 t において, ワーカー $i \in [k]$ は経験誤差を自身の保持するデータをもとに最小化
3. 各ワーカーは, w_i^{t+1} を計算し, 勾配 $\delta_i^{t+1} = w_i^{t+1} - w_G^t$ を得る
4. Server に勾配 δ_i^{t+1}を送信

Server

5. グローバルパラメータを$\delta_i^{t+1}(i \in [k])$を用いて更新し,ワーカーに更新後のパラメータ$w_G^{t+1}$を送信

Table1　記法の定義

Notation	Description
K	ワーカー数
$D_i = \{x_1^i, \cdots, x_{l_i}^i\}$	各ワーカーの取得するデータ (private)
$\|D_i\| = l_i$	各ワーカーの取得するデータ数
$l = \sum_{i \in [K]} l_i$	全ワーカーの取得するデータ数
f	学習器
$\boldsymbol{w}_G \in \mathbb{R}^n$	グローバルパラメータ

Federated Learning and Model Poisoning

Threat Model

敵対者に関して以下を仮定する.

i. インデックスmのワーカーは悪意あるワーカーである. (敵対者数を制限)

ii. 各ワーカーの保持するデータは i.i.d である. (良性な更新と悪意ある更新の判別が容易になる)

iii. 悪意あるワーカーは自身の訓練データ D_m と同様に訓練データと同一の分布からなる補助データ D_{aux} にアクセスすることができる.

Adversarial Goals

サーバで学習した識別器が, 補助データD_{aux}で意図的な誤分類を起こすこと

正解ラベル $\{y_i\}_{i=1}^r$ であるサンプル $\{x_i\}_{i=1}^r$ をもつ補助データを$\{\tau\}_{i=1}^r$ に識別させることが目標となる.

つまり,

$$\mathcal{A}(\mathcal{D}_m \cup \mathcal{D}_{\text{aux}}, \mathbf{w}_G^t) = \max_{\mathbf{w}_G^t} \sum_{i=1} \mathbf{1}[f(\mathbf{x}_i; \mathbf{w}_G^t) = \tau_i]. \quad (1)$$

Federated Learning and Model Poisoning

Stealth metrics

サーバはワーカーの敵対的な挙動を検出するような手段を取る可能性がある.

サーバはパラメータ更新の際に, 2つの情報を確認することができる.

- 検証データであるワーカーの更新によってどれだけモデルの精度が向上するかどうか
- あるワーカーの更新が他のワーカー更新と統計的にどれだけ異なるか

この2つの指標を用いても, 検知されないような攻撃を行う

Federated Learning and Model Poisoning

Accuracy checking

- 検証データで単一 の更新によってモデルの精度が向上するかどうか

時刻 t において, ワーカー i によるパラメータ $w_i^t = w_G^{t-1} + \delta_i^t$ をもつモデルの検証データセットでの精度が,ワーカー i以外で集約されたパラメータ $w_{G\setminus i}^t = w_G^{t-1} + \sum_{j\neq i} \delta_j^t$ をもつモデルの検証データセットでの精度が大きく下回っている場合, ワーカーiは異常であると判断する可能性がある. つまり, 悪意あるワーカーは, 以下を満たす必要がある. 時刻 t において, 閾値を γ_t とすると,

$$\sum_{\boldsymbol{X}_j, y_j \in D_{test}} \mathbf{1}\left[f\left(\boldsymbol{x}_j; \boldsymbol{w}_m^t\right) = y_j\right] - \mathbf{1}\left[f\left(\boldsymbol{x}_j; \boldsymbol{w}_{G\setminus m}^t\right) = y_j\right] < \gamma_t. \tag{2}$$

Federated Learning and Model Poisoning

Weight update statistics

- あるワーカーの更新が他のワーカー更新と統計的に大きく異なるか

何らかの距離尺度 $d(\cdot,\cdot)$ に従い, 互いの勾配の距離を測ることで, どれほど異なるかを測定する.

まず, 以下のように, 敵対者と正常なワーカーとの距離の範囲を求める.

$$R_m = \left[\min_{i \in [k] \setminus m} d\left(\delta_m^t, \delta_i^t\right), \max_{i \in [k] \setminus m} d\left(\delta_m^t, \delta_i^t\right) \right]. \tag{3}$$

$R^l_{\min,[k]\setminus m}, R^u_{\max,[k]\setminus m}$ をすべての正常なワーカー同士の距離の最小値と最大値とする.

次に, 閾値をκ_tとして, 敵対者が検知されないように, 以下を満たす必要がある.

$$\max\{|R_m^u - R^l_{min,[k]\setminus m}|, |R_m^l - R^u_{max,[k]\setminus m}|\} < \kappa_t. \tag{4}$$

- これは, 敵対者と他のワーカーの範囲の違いが他のワーカー同士の範囲の違いとそこまで変わらないことを保障し, R_mの長さも制御している

Strategies for Model Poisoning Attacks

Server Aggregation rule

保持するデータ数に対する重み付き平均 $w_G^t = w_G^{t-1} + \sum_{i\in[k]} \alpha_i \, \delta_i^t, where \frac{l_i}{l} = \alpha_i$

Targeted model poisoning

Objective

$$\underset{\boldsymbol{\delta}_m^t}{\operatorname{argmin}} \quad L(\{\mathbf{x}_i, \tau_i\}_{i=1}^r, \hat{\mathbf{w}}_G^t), \qquad (4)$$

※時刻$t-1$では, 他のワーカーの勾配は分からないため, w_G^tも分からない.

よって, $\widehat{w}_G^t = w_G^{t-1} + \alpha_m \delta_m^t$とし, w_G^tの推定値を用いる

※最終的に重み付き平均を行うのでα_iだけスケーリングされてしまう

よって, 敵対者はスケーリングの影響を克服するために, 勾配δ_m^tではなく,

$\lambda\delta_m^t \;\; where \;\; \lambda = \frac{1}{\alpha_m}$を送信する (Explicit Boosting).

Notation	Description
L	損失関数
x_i	敵対者が誤分類させたいサンプル
τ_i	敵対者が誤分類させたいターゲット
δ_i^t	時刻tにおけるワーカーiの勾配
w_G^t	時刻tにおけるグローバルパラメータ
$\widehat{w}_G^t$	時刻tにおけるグローバルパラメータの推定値

Strategies for Model Poisoning Attacks

Stealthy model poisoning

2つの異常検知項目 (テスト(検証)データでの精度・勾配の距離)で検出されずに
model poisoningを達成するような目的関数を設計する. 今回は, l_2ノルムを用いる.

Objective

advarsarial objective **Stealth objective**

$$\operatorname{argmin}_{\delta_m^t} \lambda \, \mathrm{L}(\{\mathrm{x}_i, \tau_i\}_{i=1}^{r}, \widehat{w}_G^t) + \mathrm{L}(\mathrm{D}_m, \widehat{w}_G^t) + \rho \left\| \delta_m^t - \hat{\delta}_{ben}^{t-1} \right\|_2 \quad (5)$$

訓練データの損失　　敵対者と正常なワーカーとの距離の比較

※時刻$t-1$では, 他のワーカーの勾配は分からないため, $w_G^t, \delta_{i\setminus m}^t$は分からない.

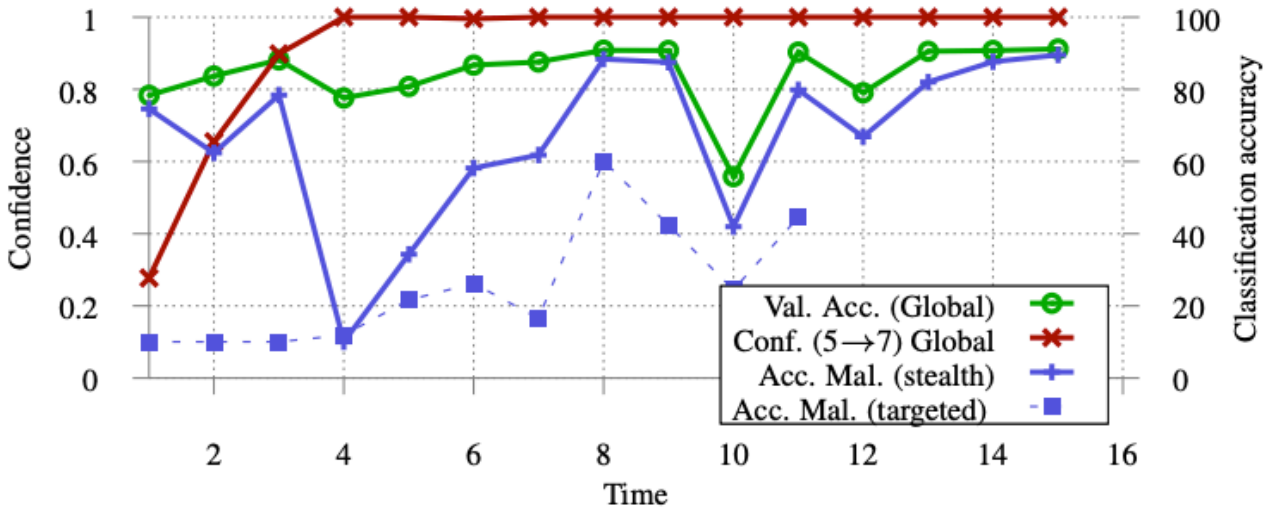
よって, $\widehat{w}_G^t = w_G^{t-1} + \alpha_m \delta_m^t$ とし, w_G^tの推定値を用いる. また, $\hat{\delta}_{\mathrm{ben}}^{t-1} = \sum_{i \in [k]\setminus m} \alpha_i \, \delta_i^{t-1}$として
前回の他のワーカーの更新を他のワーカーの更新として用いる

Strategies for Model Poisoning Attacks

Results and effect on stealth

- 高い精度を維持しつつ, 特定のクラスを誤分類させることを達成
- 検証データでの全ワーカーの精度と敵対者の精度には大きな乖離があるところが見られる

  → 全てのイテレーションで敵対者が選択されるわけではない



(a) Confidence on malicious objective and accuracy on validation data for $\mathbf{w}_G^t$. Stealth with respect to accuracy checking is also shown for both the stealthy and targeted model poisoning attacks. We use $\lambda = 10$ and $\rho = 1e^{-4}$.

Experimental setup

- Dataset : Fashion-MNIST
- Model : 3-layer CNN
- K = 10 (全ワーカー)
- k = 10 (部分集合)
- Adversarial Objective : “5” (サンダル)→”7” (スニーカー)
- 精度91% (敵対者無しのテストデータの精度)に到達するか40イテレーションに達した場合終了

Strategies for Model Poisoning Attacks

Alternating minimization for improved model poisoning

- Adversarial objectiveとStealth objectiveを交互に最小化
- 2つの値が十分小さくなるまで繰り返す

Results and effect on stealth

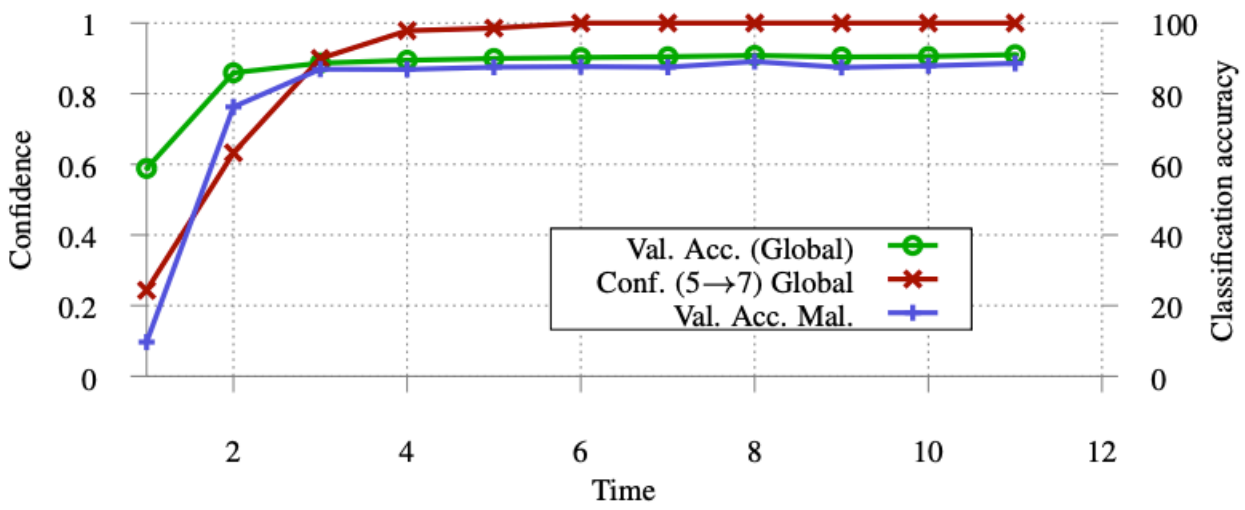


Figure 5. Alternating minimization attack with distance constraints for CNN on Fashion MNIST data. Stealth with respect to accuracy checking is also shown.

- 検証データでの全ワーカーの精度と敵対者の精度が非常に近くなった
- 初期イテレーションで大きく乖離しているように見られるので, これで成功したと言えるのかが疑問

Attacking Byzantine-resilient aggregation

Krum and Coordinate-wise median aggregation rule

- ビザンチン障害に対して堅牢な既存手法Krum (ユークリッド距離が互いに近い部分集合を選択), Coordinate-wise median (中央値を選択)に, target model poisoning,alternating minimizationを適用

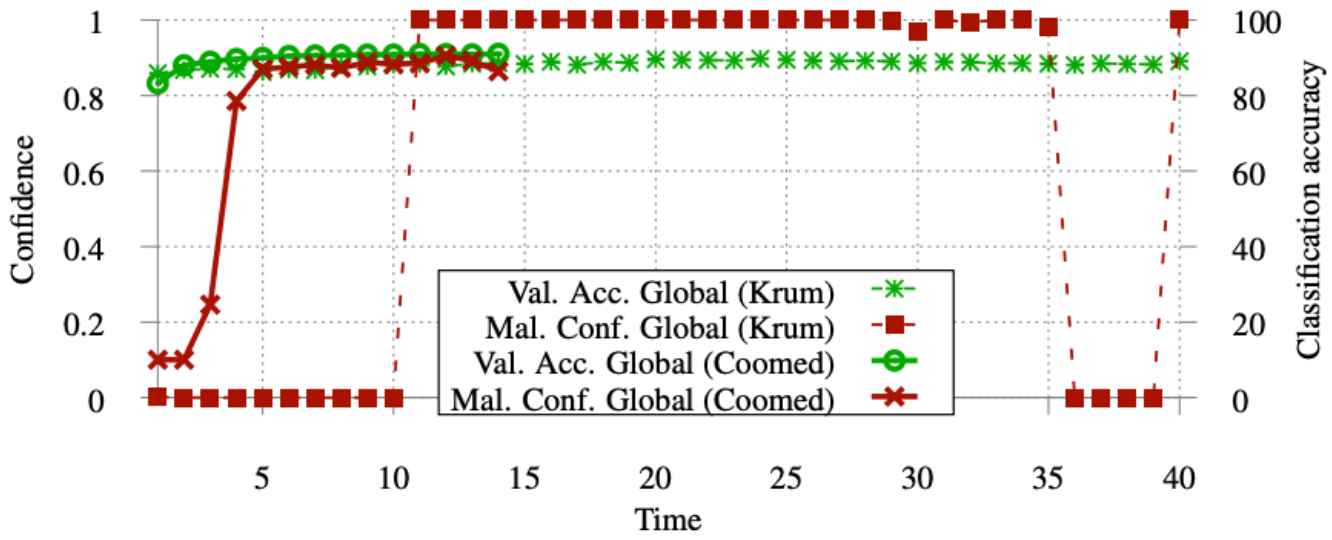


Figure 6. Model poisoning attacks with Byzantine resilient aggregation mechanisms. We use targeted model poisoning for coomed and alternating minimization for Krum.

- どちらも検証データの精度は高く, 誤分類を引き起こしていることが確認された
- Krumでは確信度がなぜ極端になってしまうのかが分からない

Discussion

Model poisoning vs. data poisoning

- 同様の評価実験 (CNN on Fashion MNIST, 10 worker)を行なった. 敵対者の訓練データにラベルを変更した誤分類させたいデータのコピーを1000個追加して, 学習を行ったが, 効果は見られなかった.

  ➡ 更新の際にスケーリングが行われるため

- 勾配をブーストすると, グローバルモデルのパフォーマンスにも影響を与えてしまう.

  ➡ Federated learningにおい, data poisoningよりもmodel poisoningの方が効果的

Conclusion

本論文では, Federated learning における, poisoning である model poisoning を定義し, それに対する脆弱性 を示した. 今後は, これを防ぐような検知方法を検討する予定である. 我々が想定した攻撃に対して, ビザンチン障害を考慮した既存手法は脆弱であることを示した. また, ここで想定 されている攻撃者に対する堅牢性はまだ示されていない.