Poisoning Attacks in Federated Learning

Presenter:
Xin Fan
2020/11/25

Outline

- Introduction to Poisoning Attacks
- Data Poisoning Attacks
- Model Poisoning Attacks
- Discussions for Poisoning Attacks in FL over the air

GEORGE MASON UNIVERSITY

Outline

- **Introduction to Poisoning Attacks**
- Data Poisoning Attacks
- Model Poisoning Attacks
- Discussions for Poisoning Attacks in FL over the air

Introduction to Poisoning Attacks

- **Passive attacks**
  - Shallow (partial information)
    - Model Extraction Attacks
    - Membership Inference Attacks
    - Model Inversion Attacks
  - Deep (original training dataset)
    - Deep Leakage
- **Active attacks**
  - Data Poisoning (targeted or untargeted)
    - Data sample tamper
    - Data label tamper
  - Model Poisoning (targeted or untargeted)

backdoor

GEORGE MASON UNIVERSITY

Outline

- Introduction to Poisoning Attacks
- **Data Poisoning Attacks**
  1. **Data evasion attack**
  2. **Data sample poisoning**
  3. **Data label poisoning**
- Model Poisoning Attacks
- Discussions for Poisoning Attacks in FL over the air

Data Poisoning Attacks

- **Data Evasion Attack**

- **Reference**
  - Goodfellow I J, Shlens J, Szegedy C. "Explaining and harnessing adversarial examples". arXiv preprint arXiv:1412.6572, 2014.

Data Poisoning Attacks

- **<u>Data Evasion Attack</u>** *v.s.* **<u>Data Poisoning Attack</u>**
  - **Data Evasion Attack**
    - Happens at **test time**
    - Perturb **a test sample** so that the model makes a classification error
  - **Data Poisoning Attack**
    - Happens at **training time**
    - Add a poison sample (**data sample poisoning**) to the training or flip a label (**data label poisoning**) .

Data Poisoning Attacks

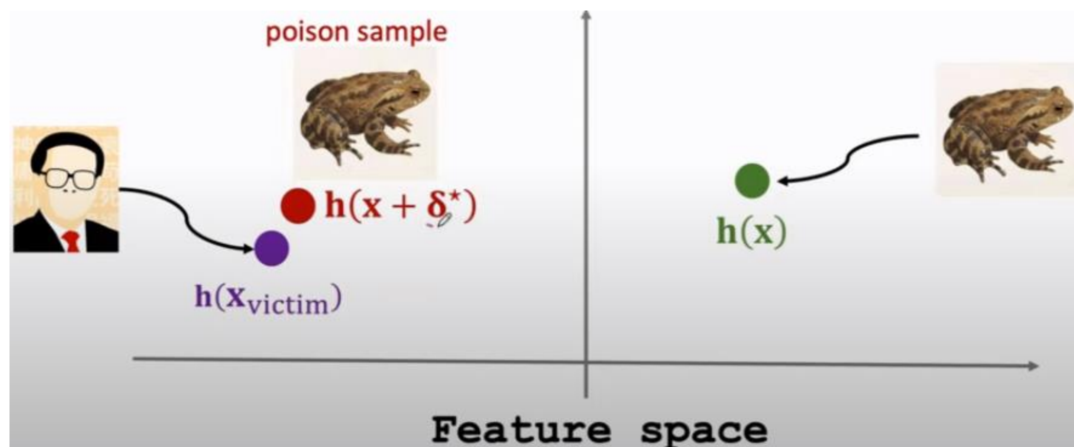
- **Data sample poisoning attack**
  - $\mathbf{x}_{\text{victim}}$: victim sample (an image not in the training set).
  - Add a perturbation $\boldsymbol{\delta}^*$ to $\mathbf{x}$ so that $\mathbf{h}(\mathbf{x}+\boldsymbol{\delta}^\star) \approx \mathbf{h}(\mathbf{x}_{\text{victim}})$
  - Optimization:

$$\boldsymbol{\delta}^\star = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} \; \left\| \mathbf{h}(\mathbf{x}+\boldsymbol{\delta}) - \mathbf{h}(\mathbf{x}_{\text{victim}}) \right\|_2^2 + \lambda \left\| \boldsymbol{\delta} \right\|_2^2 .$$

The feature vectors are similar.

The perturbation is small.

  - $\mathbf{x}$: input data
  - $\mathbf{h}(\mathbf{x})$: feature vector



Data Poisoning Attacks

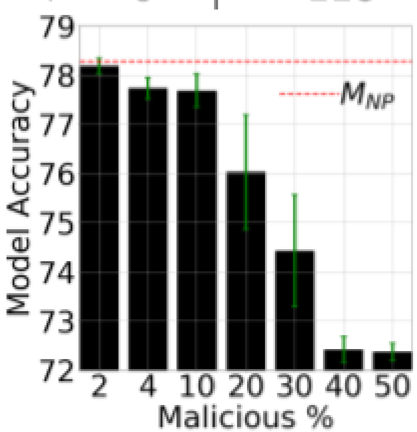
- **Data label poisoning attack**
  - m% of benign participants to poison the global model for a certain number of FL rounds
  - The final global model M has high errors for particular classes
  - Do not need to access or manipulate other participants' data or the model learning process, loss function, or server aggregation process
  - Just change a source class c_{src} to a target class c_{target} in label of the malicious participants' training datasets
- **Reference**

  - Vale Tolpegin, Stacey Truex, Mehmet Emre Gursoy, and Ling Liu. "Data Poisoning Attacks Against Federated Learning Systems." *European Symposium on Research in Computer Security*. Springer, Cham, 2020.

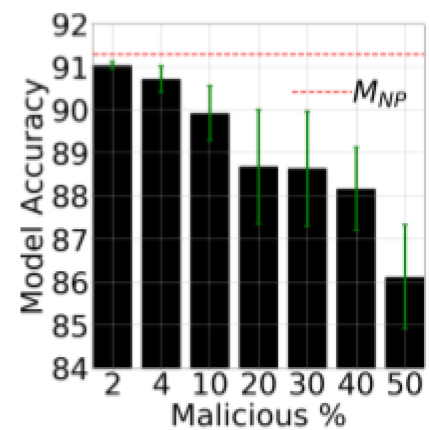
Data Poisoning Attacks

- **Data label poisoning attack**

$c_{src} \rightarrow c_{target}$	$m\_cnt^{src}_{target}$	Percentage of Malicious Participants ($m\%$)						
		2	4	10	20	30	40	50
CIFAR-10								
0 → 2	16	**1.42%**	2.93%	**10.2%**	14.1%	48.3%	**73%**	**70.5%**
1 → 9	56	0.69%	**3.75%**	6.04%	15%	36.3%	49.2%	54.7%
5 → 3	200	0%	3.21%	7.92%	**25.4%**	**49.5%**	69.2%	69.2%
Fashion-MNIST								
1 → 3	18	0.12%	0.42%	2.27%	2.41%	**40.3%**	**45.4%**	42%
4 → 6	51	**0.61%**	**7.16%**	**16%**	**29.2%**	28.7%	37.1%	**58.9%**
6 → 0	118	-1%	2.19%	7.34%	9.81%	19.9%	39%	43.4%



CIFAR-10 M^{acc}



F-MNIST M^{acc}

Data Poisoning Attacks

- **Data label poisoning attack**
  - **Defend algorithm**

Algorithm 1: Identifying Malicious Model Updates in FL

---

def `evaluate_updates(`$\mathcal{R}$: *set of vulnerable train rounds*, $\mathcal{P}$: *participant set*`):`

  $\mathcal{U} = \emptyset$

  for $r \in \mathcal{R}$ **do**

    $\mathcal{P}_r \leftarrow$ participants $\in \mathcal{P}$ queried in training round r

    $\theta_{r-1} \leftarrow$ global model parameters after training round $r-1$

    for $P_i \in \mathcal{P}_r$ **do**

      $\theta_{r,i} \leftarrow$ updated parameters after `train_DNN(`θ_{r-1}, D_i`)`

      $\theta_{\Delta,i} \leftarrow \theta_{r,i} - \theta_r$

      $\theta^{src}_{\Delta,i} \leftarrow$ parameters $\in \theta_{\Delta,i}$ connected to source class output node

      Add $\theta^{src}_{\Delta,i}$ to $\mathcal{U}$

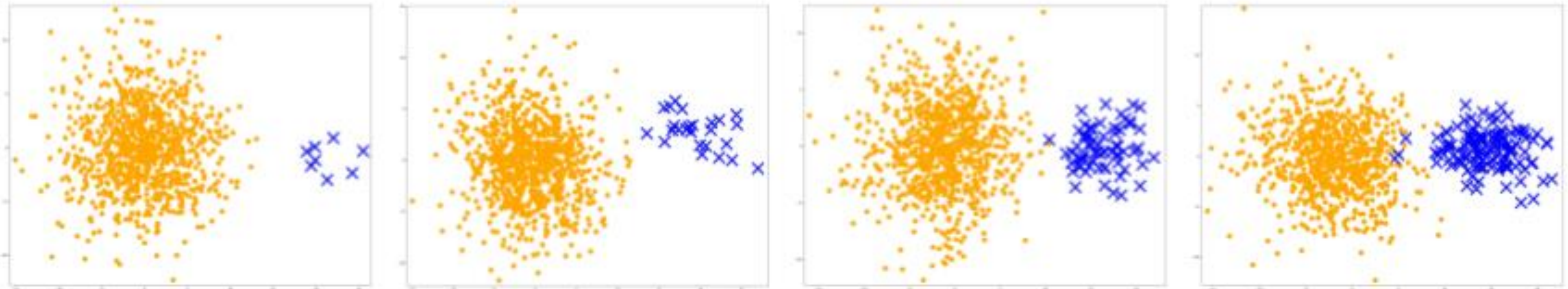
  $\mathcal{U}' \leftarrow$ `standardize(`$\mathcal{U}$`)`

  $\mathcal{U}'' \leftarrow$ `PCA(`$\mathcal{U}'$, *components=2*`)`

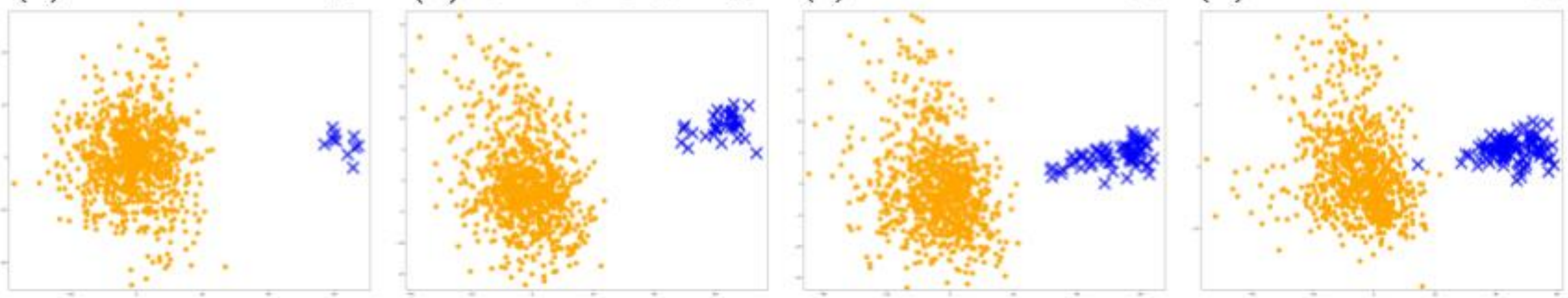
  `plot(`$\mathcal{U}''$`)`

Data Poisoning Attacks

- **Data label poisoning attack**
  - **Defend algorithm**



(a) CIFAR-10 m=2% (b) CIFAR-10 m=4% (c) CIFAR-10 m=10% (d) CIFAR-10 m=20%



(e) F-MNIST m=2% (f) F-MNIST m=4% (g) F-MNIST m =10% (h) F-MNIST m=20%

Outline

- Introduction to Poisoning Attacks
- Data Poisoning Attacks
- **Model Poisoning Attacks**
- Discussions for Poisoning Attacks in FL over the air

1. Backdoor attack
2. Defense strategy

Model Poisoning Attacks

- **Backdoor Attack (targeted model poisoning)**
  - The attacker attempts to replace the whole model by sending a deliberately carefully designed gradient
  - Backdoor the model without breaking its performance on the main task, but ensure it fails on some targeted tasks
  - Assume the attacker has a set of training samples generated from the true distribution
- **Reference**

  - Sun, Z., Kairouz, P., Suresh, A. T., & McMahan, H. B. (2019). Can you really backdoor federated learning?. *arXiv preprint arXiv:1911.07963*.

Model Poisoning Attacks

- **Backdoor Attack (targeted model poisoning)**
  - The server updates its model by aggregating the local gradients Δw_t^k's, i.e.,

$$w_{t+1} = w_t + \eta \frac{\sum_{k \in S_t} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$$

  - The attacker attempts to replace the whole model by a backdoored model w^* by sending

$$\Delta w_t^1 = \beta(w^* - w_t)$$

where $\beta = \frac{\sum_{k \in S_t} n_k}{\eta n_k}$ is a boost factor. Then we have

$$\Delta w_{t+1} = w^* + \eta \frac{\sum_{k \in S_t, k \neq 1} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$$

Model Poisoning Attacks

At the PS: $w_{t+1} = w_t + \eta \frac{\sum_{k \in S_t} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$

At the attacker: $\Delta w_t^1 = \beta(w^* - w_t)$

Derivation

$$
\begin{aligned}
w_{t+1} &= w_t + \eta \frac{\sum_{k\in S_t} n_k \Delta w_t^k}{\sum_{k\in S_t} n_k} \\
&= w_t + \eta \frac{\sum_{k\in S_t, k\neq 1} n_k \Delta w_t^k}{\sum_{k\in S_t} n_k} + \eta \frac{n_1}{\sum_{k\in S_t} n_k} \Delta w_t^1 \\
&= w_t + \eta \frac{\sum_{k\in S_t, k\neq 1} n_k \Delta w_t^k}{\sum_{k\in S_t} n_k} + \eta \frac{n_1}{\sum_{k\in S_t} n_k} \beta(w^* - w_t) \\
&= w_t + \eta \frac{\sum_{k\in S_t, k\neq 1} n_k \Delta w_t^k}{\sum_{k\in S_t} n_k} + \boxed{\eta \frac{n_1}{\sum_{k\in S_t} n_k}} \beta w^* - \boxed{\eta \frac{n_1}{\sum_{k\in S_t} n_k}} \beta w_t \\
&= \eta \frac{\sum_{k\in S_t, k\neq 1} n_k \Delta w_t^k}{\sum_{k\in S_t} n_k} + w^* \\
&\approx w^*
\end{aligned}
$$

$$\eta \frac{n_1}{\sum_{k\in S_t} n_k} \beta = 1 \Leftrightarrow \beta = \frac{\sum_{k\in S_t} n_k}{\eta n_1}$$

where $\beta = \frac{\sum_{k \in S_t} n_k}{\eta n_k}$ is a boost factor. Then we have

$$\Delta w_{t+1} = w^* + \eta \frac{\sum_{k \in S_t, k \neq 1} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$$

Original

Model Poisoning Attacks

- **Backdoor Attack (targeted model poisoning)**
  - The server updates its model by aggregating the local gradients Δw_t^k's, i.e.,

$$w_{t+1} = w_t + \eta \frac{\sum_{k \in S_t} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$$

  - The attacker attempts to replace the whole model by a backdoored model w^* by sending

$$\Delta w_t^1 = \beta(w^* - w_t)$$

where $\beta = \frac{\sum_{k \in S_t} n_k}{\eta n_1}$ is a boost factor. Then we have

Wrong in their original paper! $\cancel{\Delta} w_{t+1} = w^* + \eta \frac{\sum_{k \in S_t, k \neq 1} n_k \Delta w_t^k}{\sum_{k \in S_t} n_k}$

Model Poisoning Attacks

- **Defense strategy**
  - **Norm thresholding of updates**
    - **The following norm-clipping approach:**

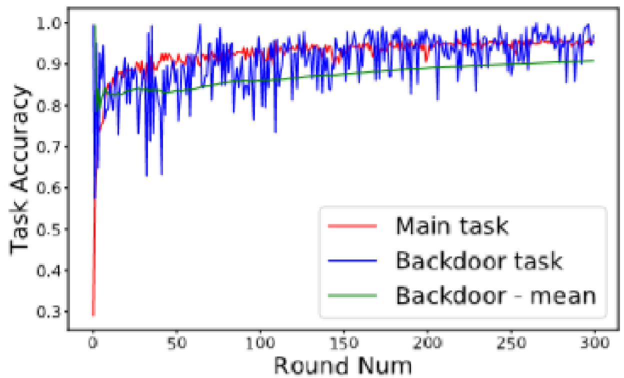
$$\Delta w_{t+1} = \sum_{k \in S_t} \frac{\Delta w_{t+1}^k}{\max(1, \|\Delta w_{t+1}^k\|_2 / M)}$$

  - **(Weak) differential privacy**
    - **by first clipping updates (as above) and then adding Gaussian noise.**

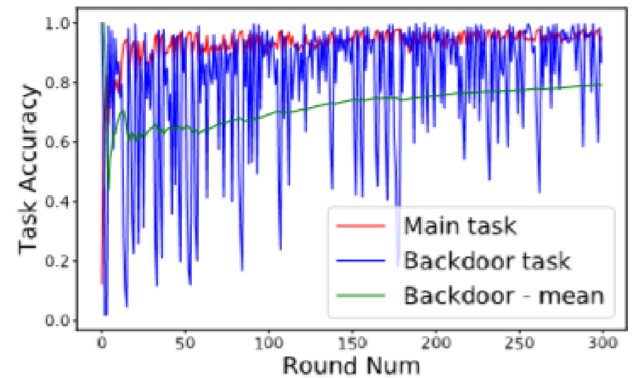
small

Model Poisoning Attacks

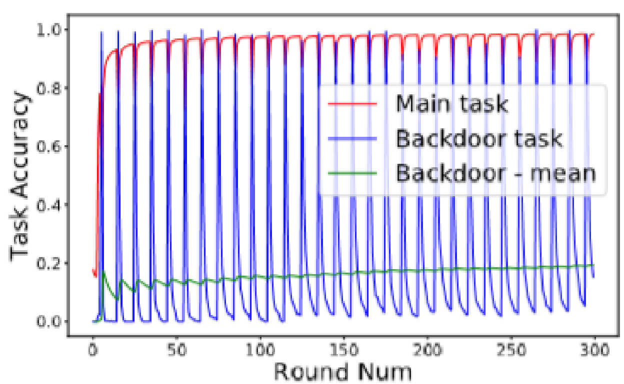
- **Experiments**



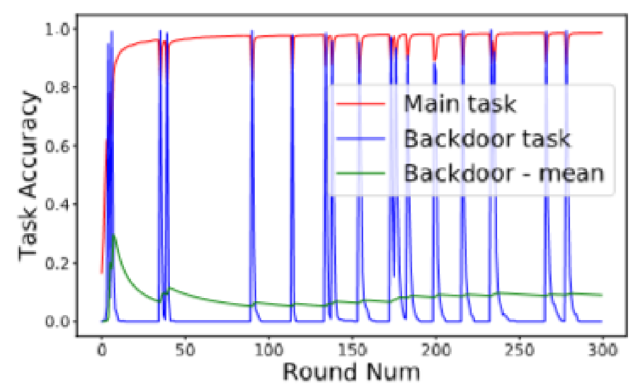
(a) Attack frequency = 1 ($\epsilon = 3.3\%$)



(b) Number of attackers = 113 ($\epsilon = 3.3\%$)



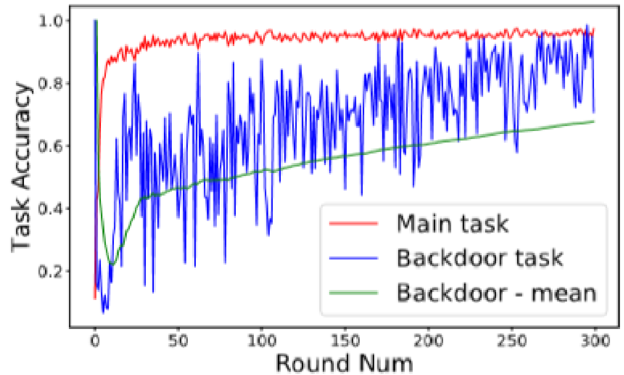
(c) Attack frequency = 1/10 ($\epsilon = 0.33\%$)



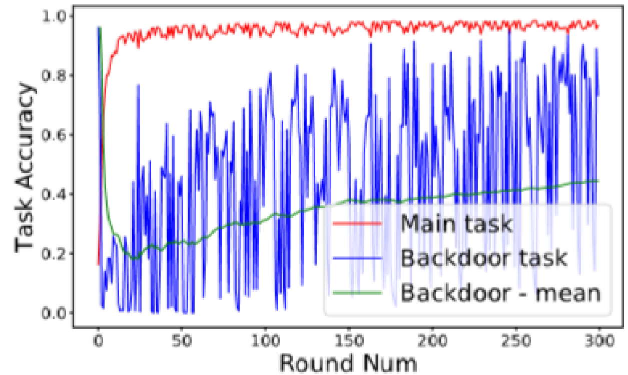
(d) Number of attackers = 11 ($\epsilon = 0.33\%$)

Model Poisoning Attacks

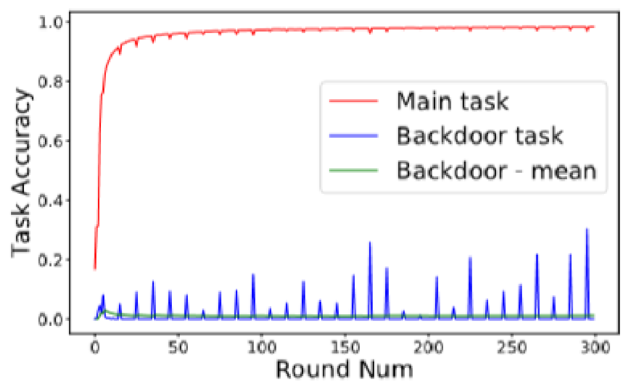
- **Experiments**



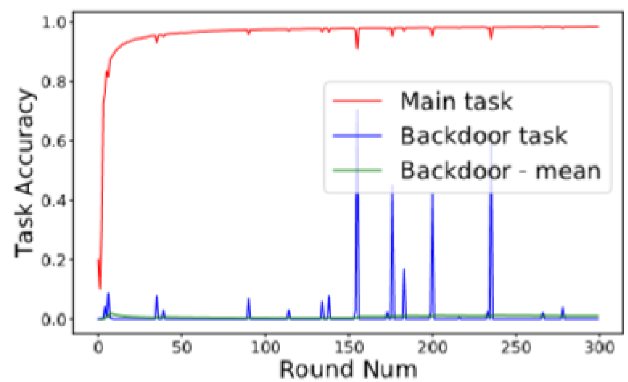
(a) Attack frequency = 1 (ϵ = 3.3%)



(b) Number of attackers = 113 (ϵ = 3.3%)



(c) Attack frequency = 1/10 (ϵ = 0.33%)



(d) Number of attackers = 11 (ϵ = 0.33%)

Outline

- Introduction to Poisoning Attacks
- Data Poisoning Attacks
- Model Poisoning Attacks
- **Discussions for Poisoning Attacks in FL over the air**

Discussions for Poisoning Attacks in FL over the air

- **Can these active or passive attacks be applied in FL over the air?**
  - Deep leakage can not be applied
  - Poisoning attacks can be applied, and it is more difficult to defend them than that in non-over-the air based FL
- **How to defend them?**
  - **SignSGD:** a voting mechanism (for the scenarios that the attacker portion is lower than 50%)
  - **Random worker selection:** a opportunistic defense mechanism

GEORGE MASON UNIVERSITY

Questions?

Thanks!