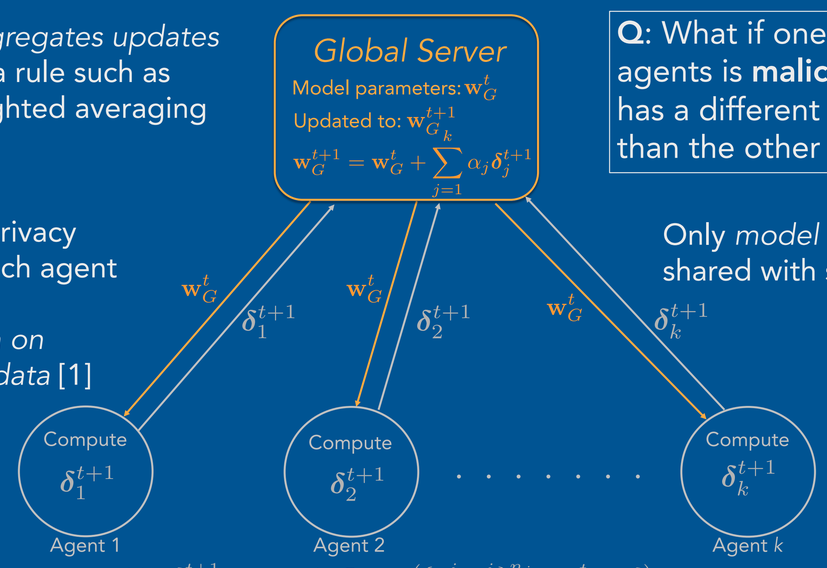
Model Poisoning Attacks on Federated Learning

Arjun Nitin Bhagoji [1], Supriyo Chakraborty [2],
Prateek Mittal [1] and Seraphin Calo [2]
[1] Princeton University [2] I.B.M. Research

Federated Learning

Server *aggregates updates* based on a rule such as linear weighted averaging (shown)

Guided by privacy concerns, each agent performs *computation on locally held data* [1]



Q: What if one of the agents is **malicious**, i.e. has a different objective than the other agents?

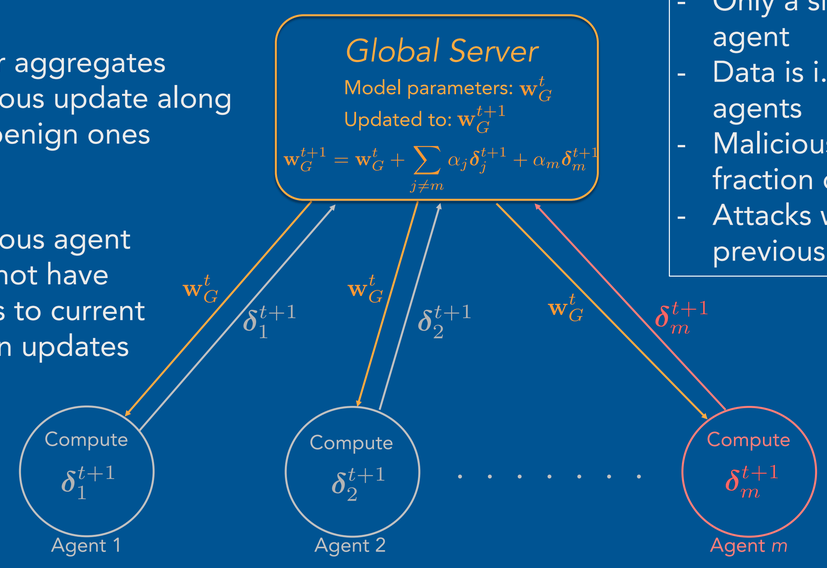
Only *model updates* are shared with server

$$\forall j,\ \boldsymbol{\delta}_j^{t+1} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{train}}\left(\{\mathbf{x}_j^i, y_j^i\}_{i=1}^{n_j}; \mathbf{w}_G^t + \boldsymbol{\delta}\right)$$

Threat Model

Server aggregates malicious update along with benign ones

Malicious agent does not have access to current benign updates



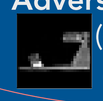
- Only a single malicious agent
- Data is i.i.d. among agents
- Malicious agent has a fraction of benign data
- Attacks with respect to previous global state

Malicious agent returns update computed to achieve targeted misclassification of a few samples

$$\forall j \neq m,\ \boldsymbol{\delta}_j^{t+1} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{train}}\left(\{\mathbf{x}_j^i, y_j^i\}_{i=1}^{n_j}; \mathbf{w}_G^t + \boldsymbol{\delta}\right) \qquad \boldsymbol{\delta}_m^{t+1} = \mathcal{A}\left(\{\mathbf{x}_m^i, y_m^i\}_{i=1}^{n_m}, \{\mathbf{x}^l, T^l\}_{l=1}^{n_{\text{mal}}}; \mathbf{w}_G^t + \boldsymbol{\delta}\right)$$

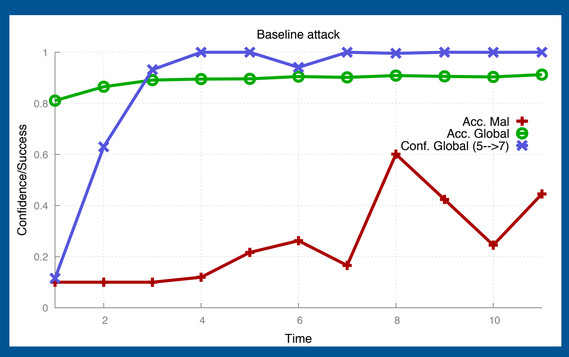
Attack Strategies for Model Poisoning

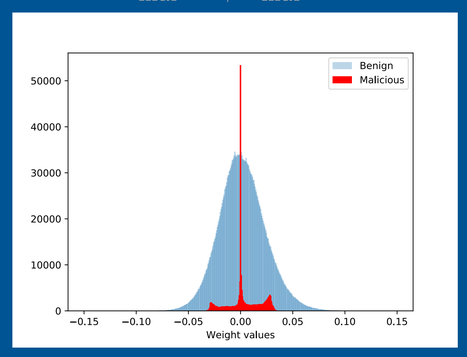
Dataset: Fashion MNIST [2]
Model: CNN with 91.7% test set accuracy

Adversarial goal is to ensure  (sandal, class 5) is classified as a sneaker (class 7)

Baseline attack (explicit boosting)

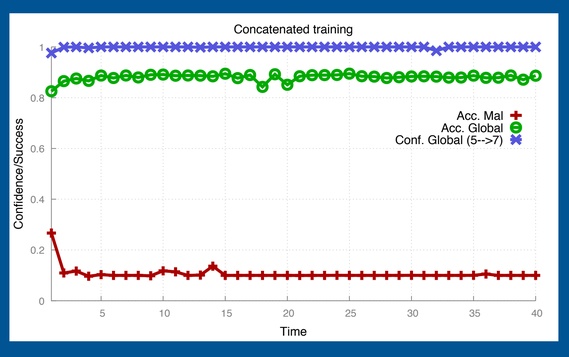
$$\boldsymbol{\delta}_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{mal}}\left(\{\mathbf{x}^l, T^l\}_{l=1}^{n_{\text{mal}}}; \mathbf{w}_G + \boldsymbol{\delta}\right)$$
$$\boldsymbol{\delta}_{\text{mal}} \rightarrow \beta \boldsymbol{\delta}_{\text{mal}}$$

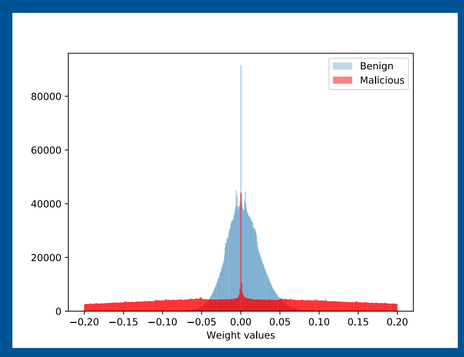




Concatenated training

$$\boldsymbol{\delta}_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{mal}}\left(\{\mathbf{x}^l, T^l\}_{l=1}^{n_{\text{mal}}}, \{\mathbf{x}_m^i, y_m^i\}_{i=1}^{n_m}; \mathbf{w}_G + \boldsymbol{\delta}\right)$$
$$\boldsymbol{\delta}_{\text{mal}} \rightarrow \beta \boldsymbol{\delta}_{\text{mal}}$$



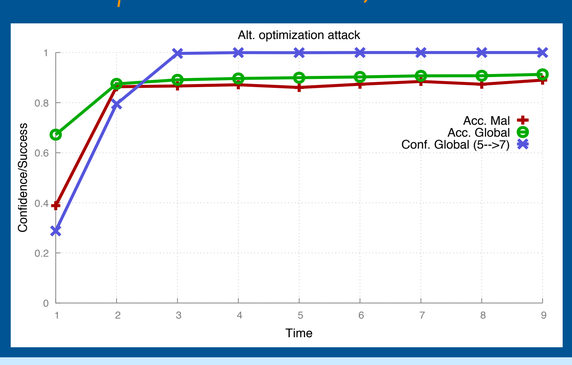


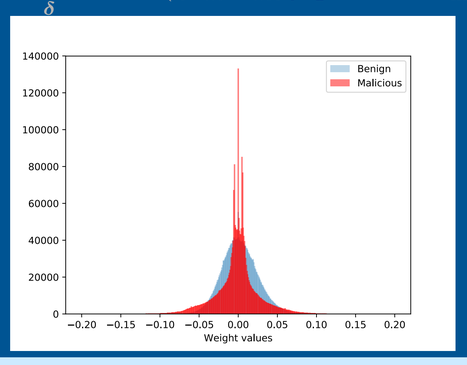
Alternating minimization (switching between benign and malicious updates for stealth)

$$\boldsymbol{\delta}'_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{mal}}\left(\{\mathbf{x}^l, T^l\}_{l=1}^{n_{\text{mal}}}; \mathbf{w}_G + \boldsymbol{\delta}\right)$$

Repeat:

$$\boldsymbol{\delta}'_{\text{mal}} \rightarrow \beta \boldsymbol{\delta}'_{\text{mal}}$$
$$\boldsymbol{\delta}''_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{ben}}\left(\{\mathbf{x}_m^i, y_m^i\}_{i=1}^{n_m}; \mathbf{w}_G + \beta \boldsymbol{\delta}'_{\text{mal}} + \boldsymbol{\delta}\right)$$



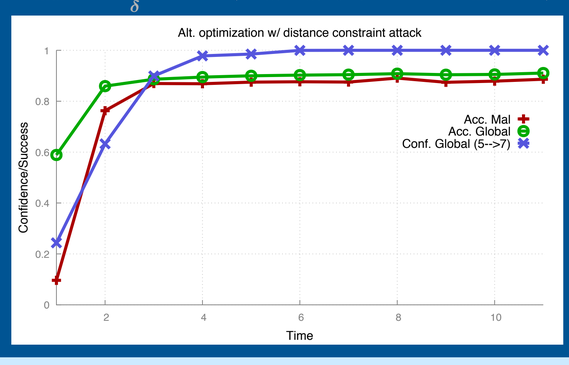


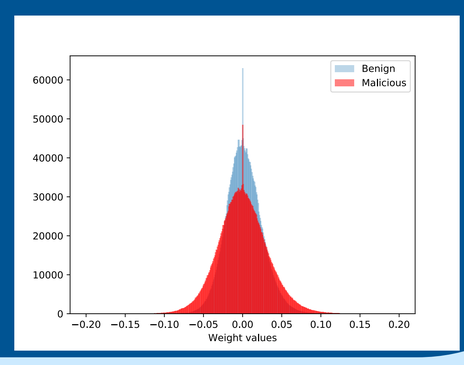
Alternating minimization with distance constraints

$$\boldsymbol{\delta}'_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{mal}}\left(\{\mathbf{x}^l, T^l\}_{l=1}^{n_{\text{mal}}}; \mathbf{w}_G + \boldsymbol{\delta}\right)$$

Repeat:

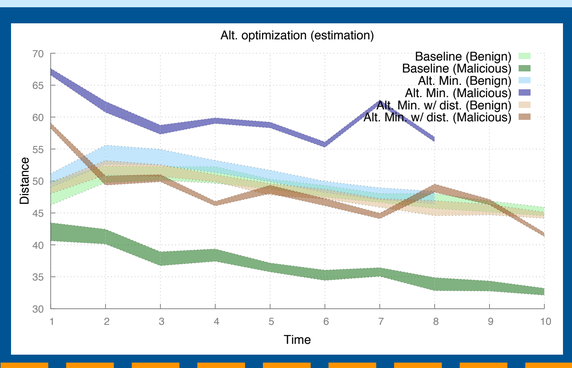
$$\boldsymbol{\delta}'_{\text{mal}} \rightarrow \beta \boldsymbol{\delta}'_{\text{mal}}$$
$$\boldsymbol{\delta}''_{\text{mal}} = \underset{\boldsymbol{\delta}}{\operatorname{argmin}} L_{\text{ben}}\left(\{\mathbf{x}_m^i, y_m^i\}_{i=1}^{n_m}; \mathbf{w}_G + \beta \boldsymbol{\delta}'_{\text{mal}} + \boldsymbol{\delta}\right) + \rho \|\boldsymbol{\delta} - \boldsymbol{\delta}_{\text{cons}}\|_2^2$$





Attack stealth measure: distance spread

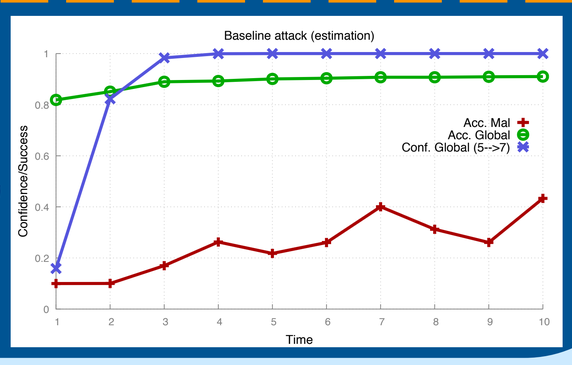
For each strategy, we show the spread of L_2 distances between all the benign agents and between the malicious agent and the benign agents.



Estimation to improve attacks

Pre-optimization correction with previous step estimate of benign agents' effects

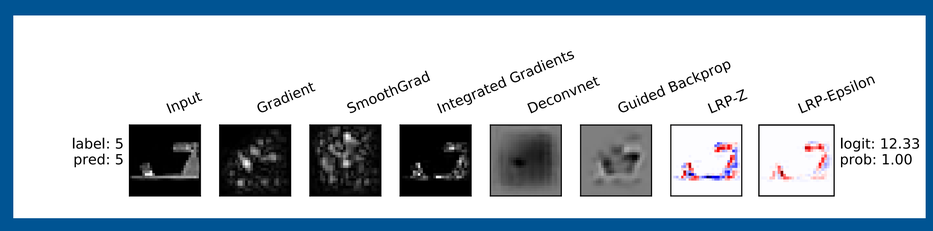
$$\hat{\mathbf{w}}_G^t = \hat{\mathbf{w}}_G^{t-1} + \hat{\boldsymbol{\delta}}_{[k]\setminus m} + \alpha_m \boldsymbol{\delta}_m^t$$
$$\hat{\boldsymbol{\delta}}_{[k]\setminus m} = \boldsymbol{\delta}_{[k]\setminus m}^{t-1}$$



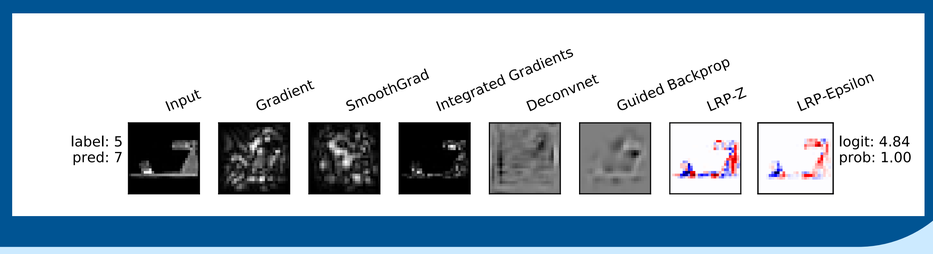
Interpreting Poisoned Models

- Interpretability techniques provide insights into the internal feature representations and working of a neural network
- Used a suite of these techniques [3] to understand decisions of poisoned models

Global model trained using only benign agents



Global model trained with one malicious model and the rest benign



Conclusion

Our attacks in this paper demonstrate that federated learning in its basic form is very vulnerable to model poisoning adversaries. While detection mechanisms can make these attacks more challenging, these can be overcome, demonstrating that multi-party machine learning algorithms robust to attackers of the type considered here must be developed.

References

[1] McMahan et al., *Communication-Efficient Learning of Deep Networks from Decentralized Data*, AISTATS 2017
[2] Xiao et al., *Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms*, arXiv preprint arXiv:1708.07747, 2017
[3] Alber et al., *iNNvestigate neural networks!*, arXiv preprint arXiv:1808.04260, 2018