Robust Learning for Data Poisoning Attacks

Yunjuan Wang, Poorya Mianjy, Raman Arora
Johns Hopkins University

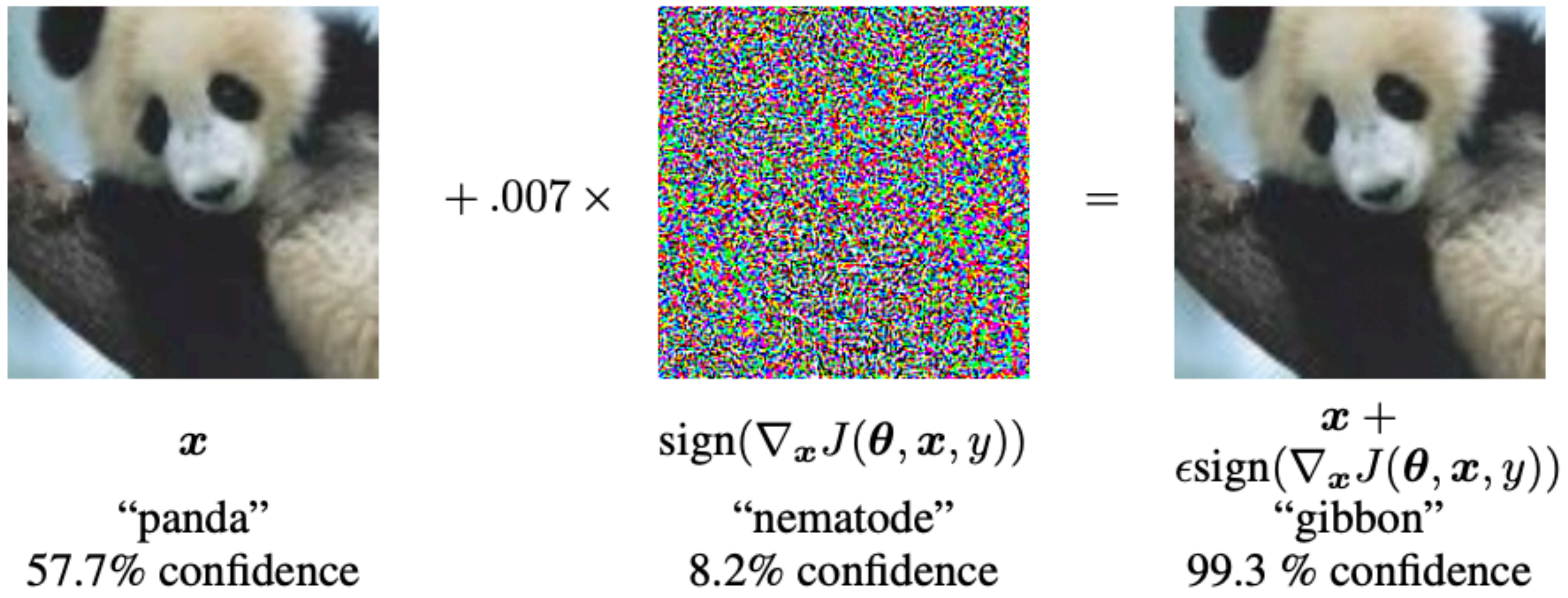
ICML 2021

Background

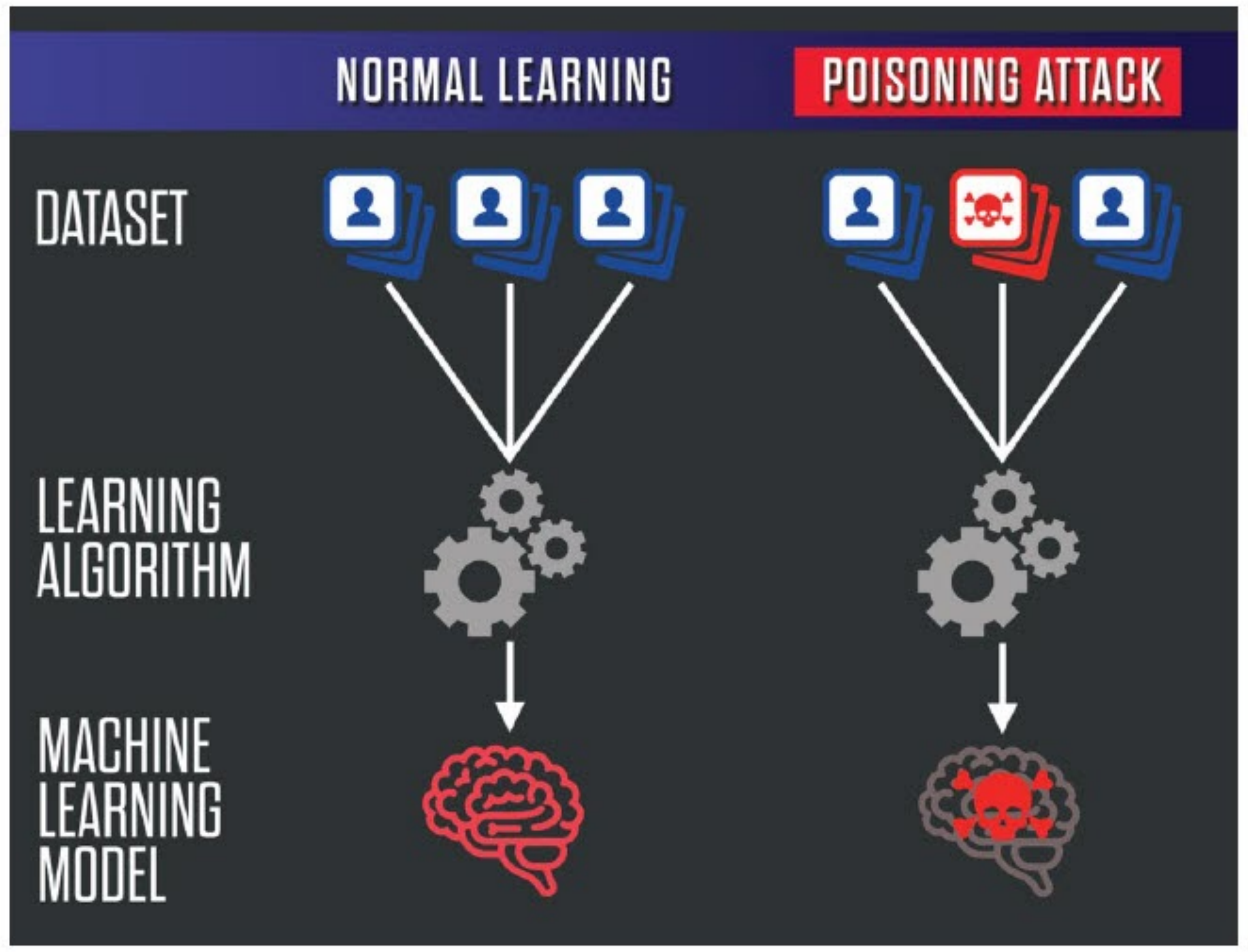
- ML systems are fragile, susceptible to attacks.

Types of attacks

- Inference-time
- Data poisoning



[GoodFellow et al., 2014]



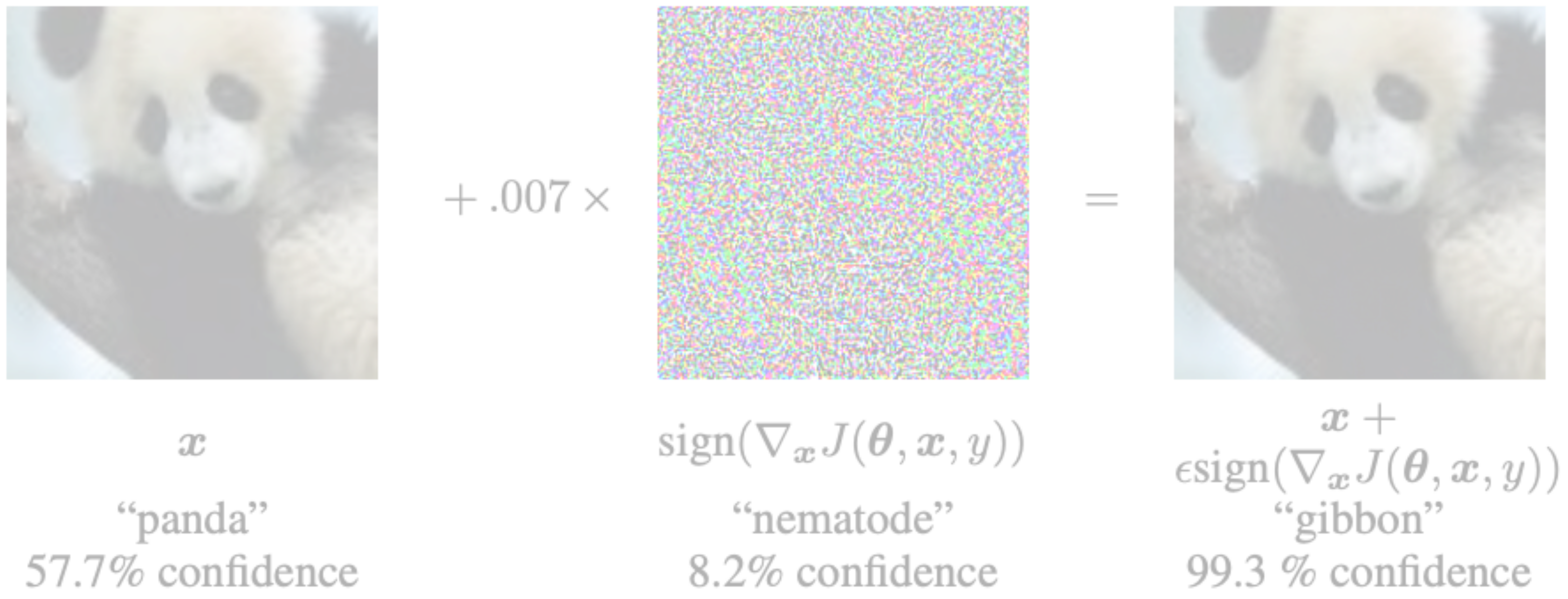
[informatiomatters.net]

Background

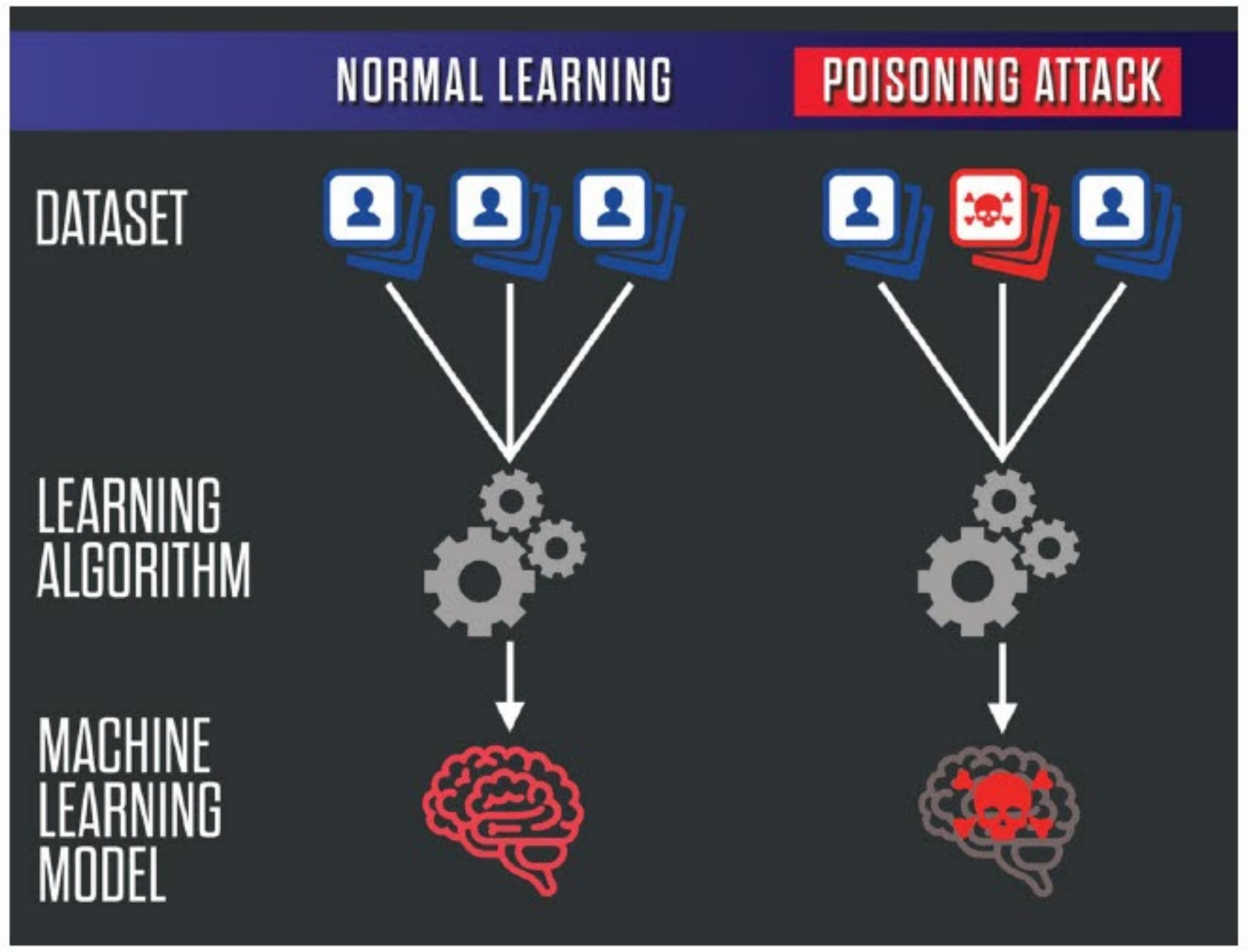
- ML systems are fragile, susceptible to attacks.

Types of attacks

Inference-time | Data poisoning



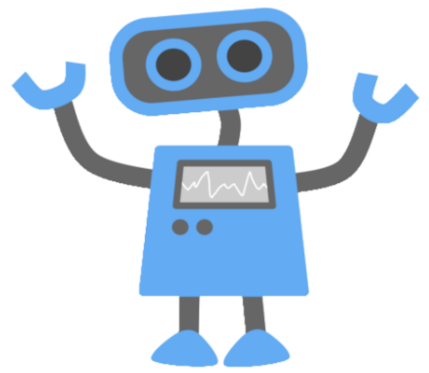
[GoodFellow et al., 2014]



[informatiomatters.net]

- In this work, we focus on data poisoning attacks.

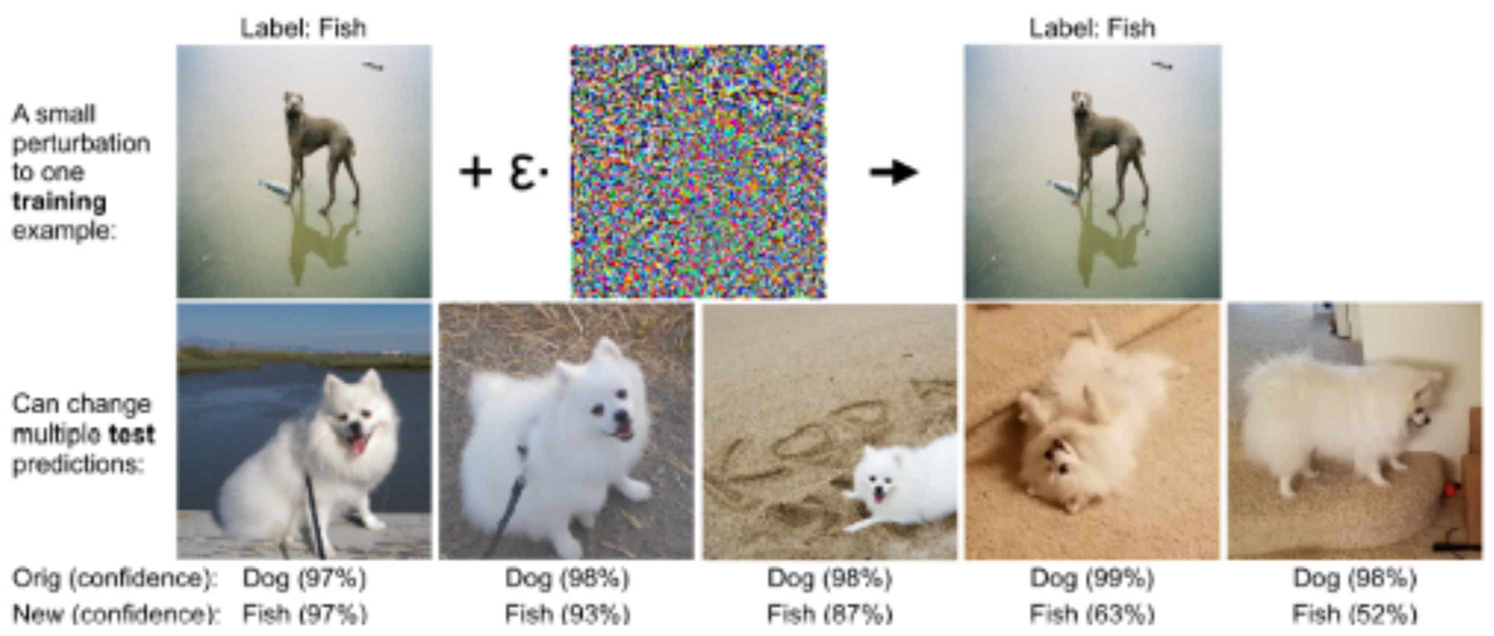
Data poisoning attack

- The adversary manipulates the training data.
- $\mathcal{S} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{n}$  $\tilde{\mathcal{S}} = \{(\tilde{\mathrm{x}}_i, \tilde{y}_i)\}_{i=1}^{n}$ 

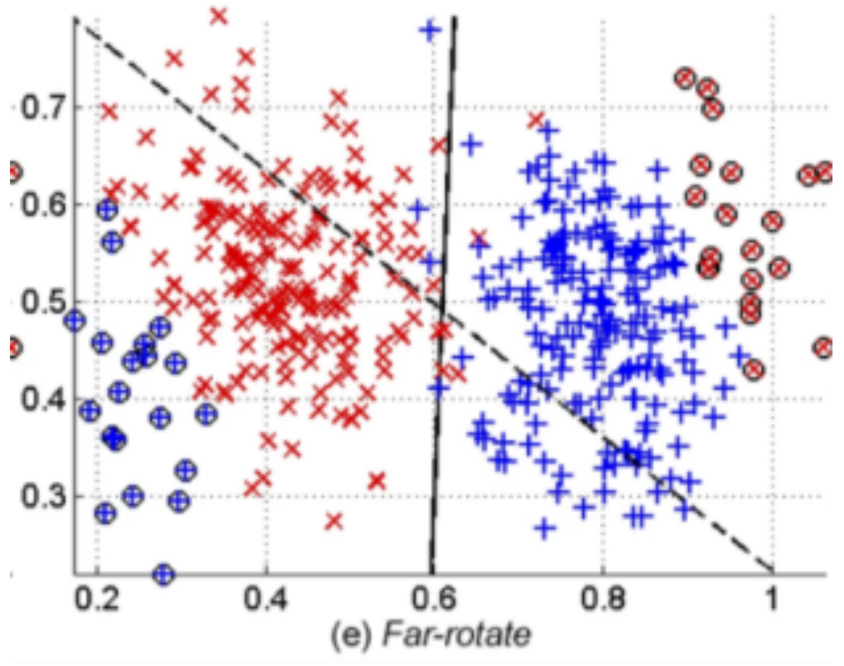
Data poisoning attacks

- Backdoor attack
- Clean label attack
- Label flip attack

[Gu et al., 2017]

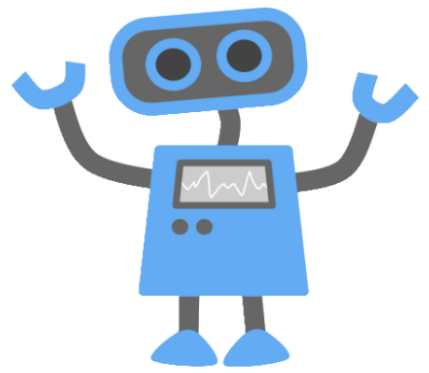


[Koh & Liang, 2017]



[Chan et al., 2021]

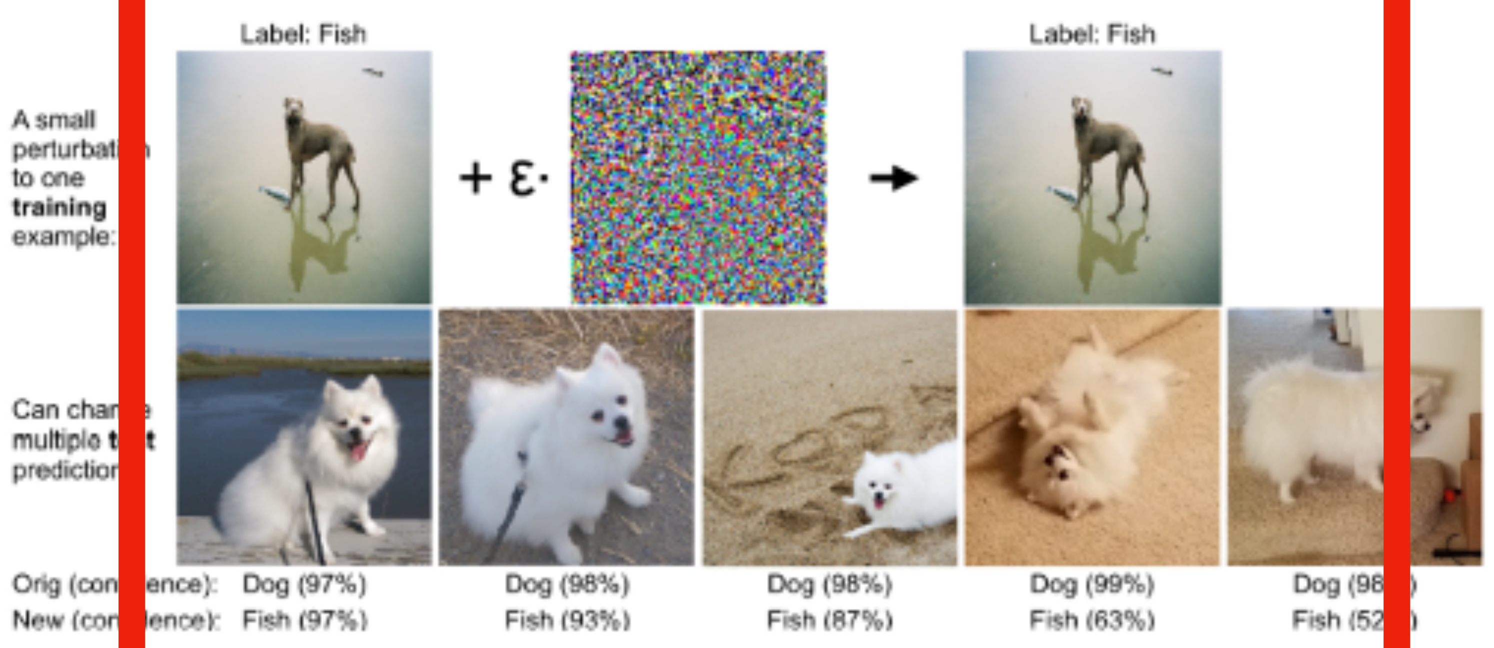
Data poisoning attack

- The adversary manipulates the training data.
- $\mathcal{S} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{n}$  $\tilde{\mathcal{S}} = \{(\tilde{\mathrm{x}}_i, \tilde{y}_i)\}_{i=1}^{n}$ 

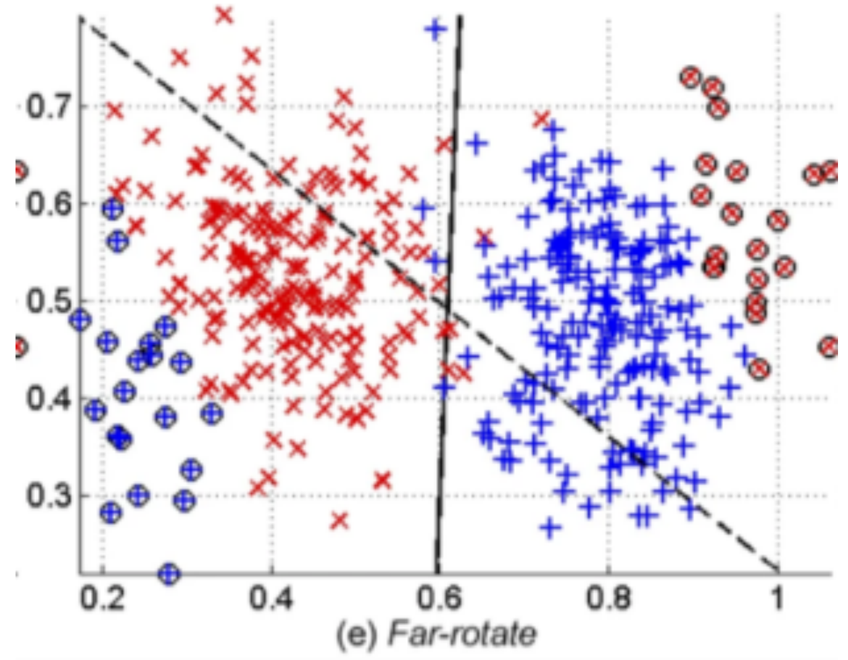
Data poisoning attacks

Backdoor attack | Clean label attack | Label flip attack

[Gu et al., 2017]



[Koh & Liang, 2017]



[Chan et al., 2021]

We focus on the latter two cases: $\tilde{\mathrm{x}}_i = \mathrm{x}_i + \delta_i, \tilde{y}_i = y_i$; $\tilde{\mathrm{x}}_i = \mathrm{x}_i, \tilde{y}_i = -y_i$ w.p. β

Convex learning problem

Goal: solve the stochastic optimization problem

$$\min_{\mathrm{w} \in \mathsf{W}} F(\mathrm{w}) := \mathbb{E}_{(\mathrm{x},y)\sim\mathscr{D}}[\ell(yf(\mathrm{x};\mathrm{w}))],$$

where W is a convex set, ℓ is convex in w.

Convex learning problem

Goal: solve the stochastic optimization problem

$$\min_{\mathrm{w} \in \mathsf{W}} F(\mathrm{w}) := \mathbb{E}_{(\mathrm{x},y)\sim\mathscr{D}}[\ell(yf(\mathrm{x};\mathrm{w}))],$$

where W is a convex set, ℓ is convex in w.

Standard approach is to use SGD, where the learner takes w and gets access to a first order stochastic oracle for $\hat{\mathrm{g}}(\mathrm{w}) \in \partial F(\mathrm{w})$.

Convex learning problem

Goal: solve the stochastic optimization problem

$$\min_{\mathrm{w} \in \mathsf{W}} F(\mathrm{w}) := \mathbb{E}_{(\mathrm{x},y)\sim\mathscr{D}}[\ell(yf(\mathrm{x};\mathrm{w}))],$$

where W is a convex set, ℓ is convex in w.

Standard approach is to use SGD, where the learner takes w and gets access to a first order stochastic oracle for $\hat{\mathrm{g}}(\mathrm{w}) \in \partial F(\mathrm{w})$.

Observation: data poisoning attacks (δ_i) can be viewed as oracle poisoning attacks (ζ_i).

- $\delta_i = \tilde{\mathrm{x}}_i - \mathrm{x}_i$.
- $\zeta_i = \tilde{\mathrm{g}}(\mathrm{w}_i) - \hat{\mathrm{g}}(\mathrm{w}_i)$.

Convex learning problem

Goal: solve the stochastic optimization problem

$$\min_{\mathrm{w} \in \mathsf{W}} F(\mathrm{w}) := \mathbb{E}_{(\mathrm{x},y)\sim\mathcal{D}}[\ell(yf(\mathrm{x};\mathrm{w}))],$$

where W is a convex set, ℓ is convex in w.

Main Result: Excess risk bound for clean label attacks:

$$\mathbb{E}[F(\bar{\mathrm{w}})] - F(\mathrm{w}_*) \leq O(\frac{1}{\sqrt{n}} + \frac{\sum_{i<n} \|\zeta_i\|}{n})$$

Convex learning problem

Goal: solve the stochastic optimization problem

$$\min_{\mathrm{w} \in \mathsf{W}} F(\mathrm{w}) := \mathbb{E}_{(\mathrm{x},y)\sim\mathcal{D}}[\ell(yf(\mathrm{x};\mathrm{w}))],$$

where W is a convex set, ℓ is convex in w.

Main Result: Excess risk bound for clean label attacks:

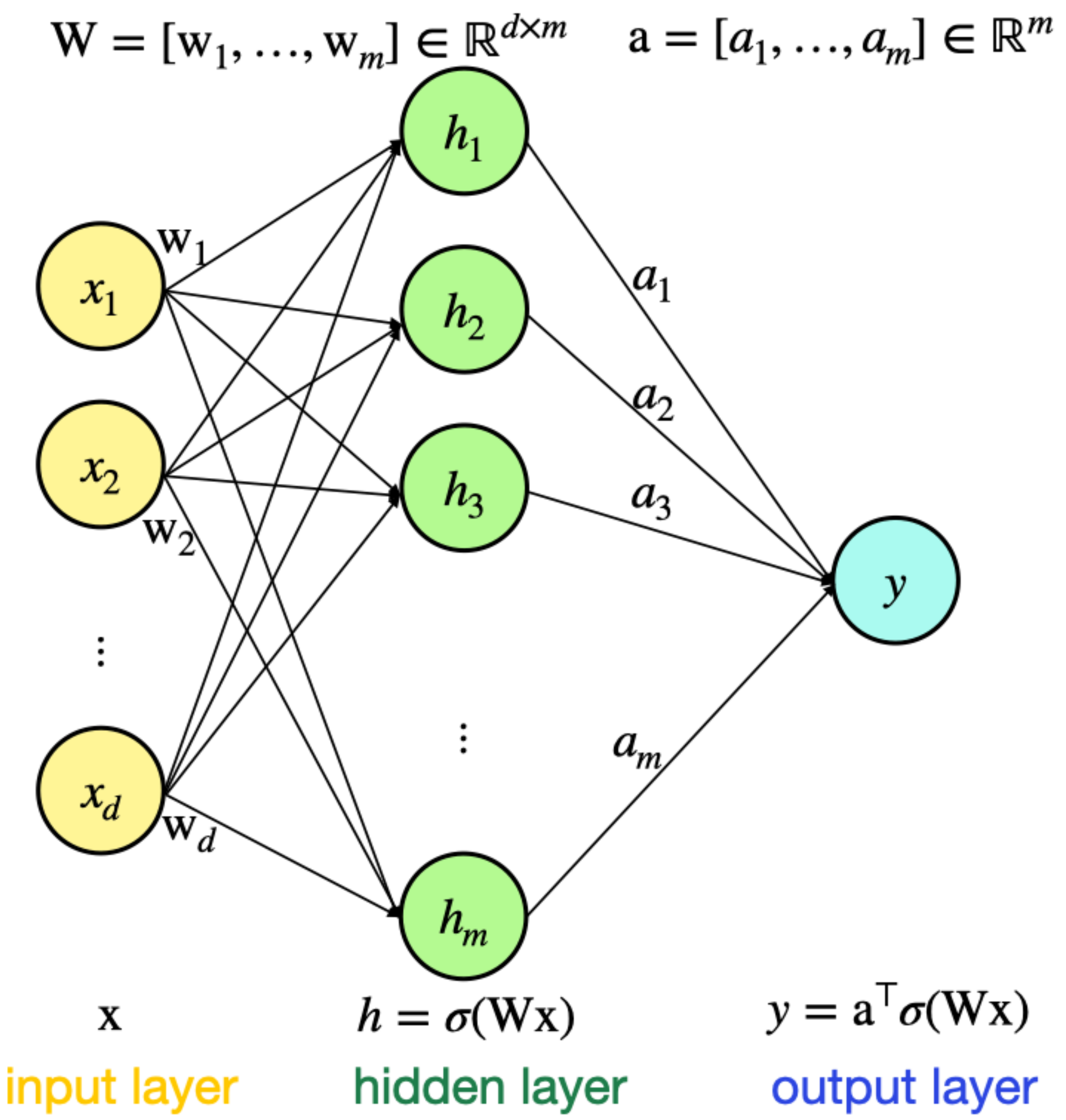
$$\mathbb{E}[F(\bar{\mathrm{w}})] - F(\mathrm{w}_*) \leq O(\frac{1}{\sqrt{n}} + \frac{\sum_{i<n} \|\zeta_i\|}{n})$$

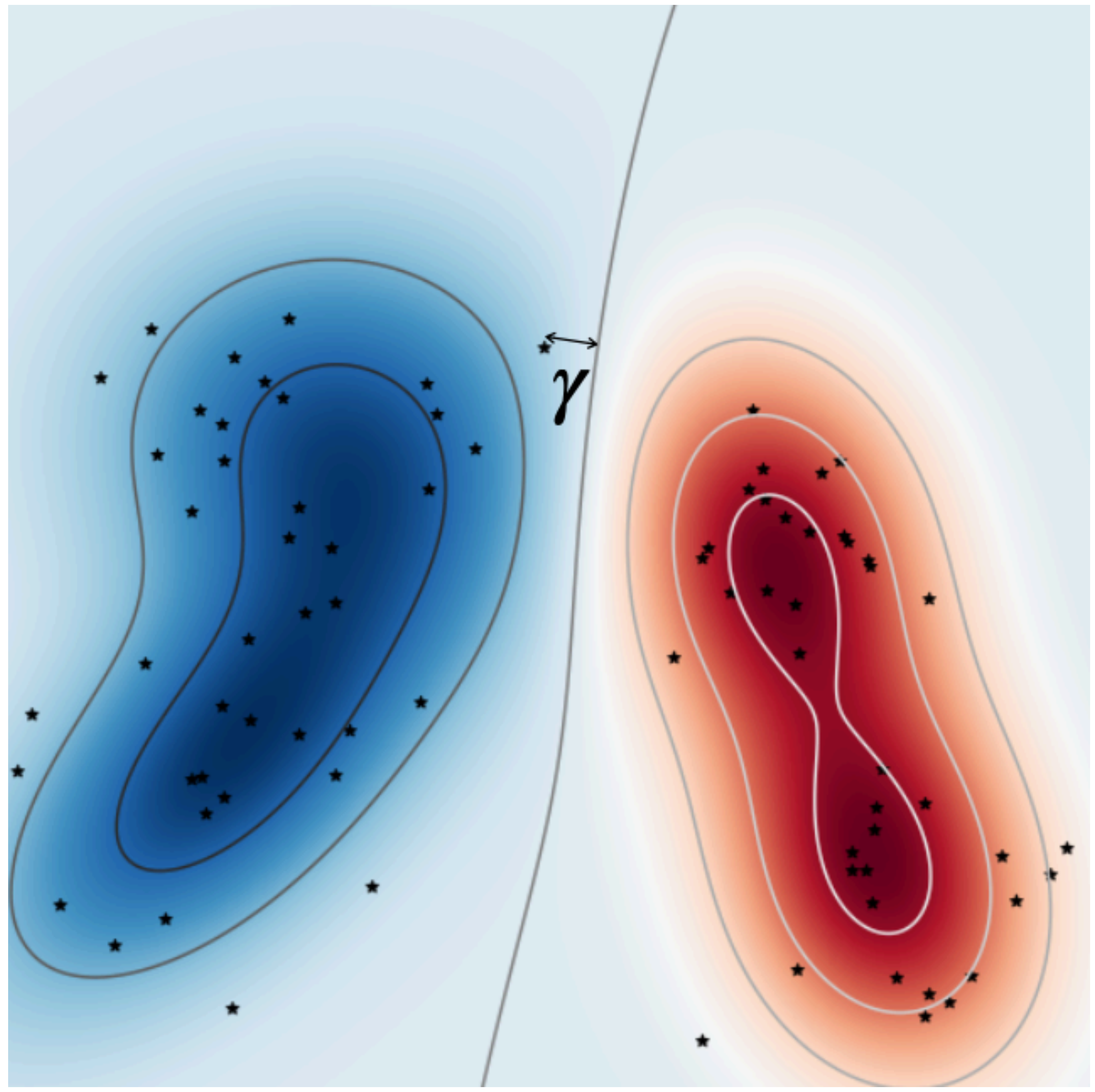
Remark: 1. $\sum_{i<n} \|\zeta_i\| = \mathcal{O}(\sqrt{n})$ gives *no significant statistical overhead.*

2. The above upper bound is *tight* in an information-theoretic sense (see paper for a lower bound).

Two-layer neural networks

- A two-layer ReLU net parameterized by (a, W),
$$f(\mathrm{x}; \mathrm{a}, \mathrm{W}) := \frac{1}{\sqrt{m}} \sum_{s=1}^{m} a_s \sigma(\mathrm{w}_s^\top \mathrm{x}),\ \sigma(z) \text{ is ReLU.}$$
- Trained by online SGD using logistic loss.
- Goal: minimize $L(\mathrm{W}) := \mathbb{P}_{(\mathrm{x},y)\sim\mathscr{D}}(yf(\mathrm{x}; \mathrm{a}, \mathrm{W}) < 0)$.





Assumption: The data distribution is separable by a positive margin γ in the reproducing kernel Hilbert space induced by the gradient of the infinite-width network at initialization, [(Du et al., 2018), (Ji & Telgarsky, 2019)].

Main Result

Regime A (clean label attacks)

Theorem: With probability at least $1-\delta$, we show the following for the iterates of SGD:

$$\frac{1}{n}\sum_{i<n} L(\mathrm{W}_i) \lesssim \frac{\ln^2(\sqrt{n}/4) + \ln(24n/\delta)}{\sqrt{n}\gamma^2}$$

provided that $B \leq \tilde{\mathcal{O}}(\gamma/\sqrt{d})$, $\tilde{\mathcal{O}}(\frac{1}{\gamma^8}) \leq m \leq \tilde{\mathcal{O}}(\frac{n}{\gamma^4 S^2})$. $\tilde{\mathcal{O}}$ hides poly-logarithmic dependence on n.

Remark: 1. B is per-sample perturbation; S is overall perturbation; m is the network width.

2. $S \lesssim \gamma^2\sqrt{n}$ to allow a *non-empty width range*.

3. Theorem implies SGD can handle *large* per-sample perturbation, as long as overall perturbation is *small*.

For other regimes like small per-sample perturbation with large overall perturbation setting (**Regime B**), and label flip attack (**Regime C**), check our paper for details.

Experiments

- Main takeaway: networks that are extremely over-parameterized are more susceptible to attacks.

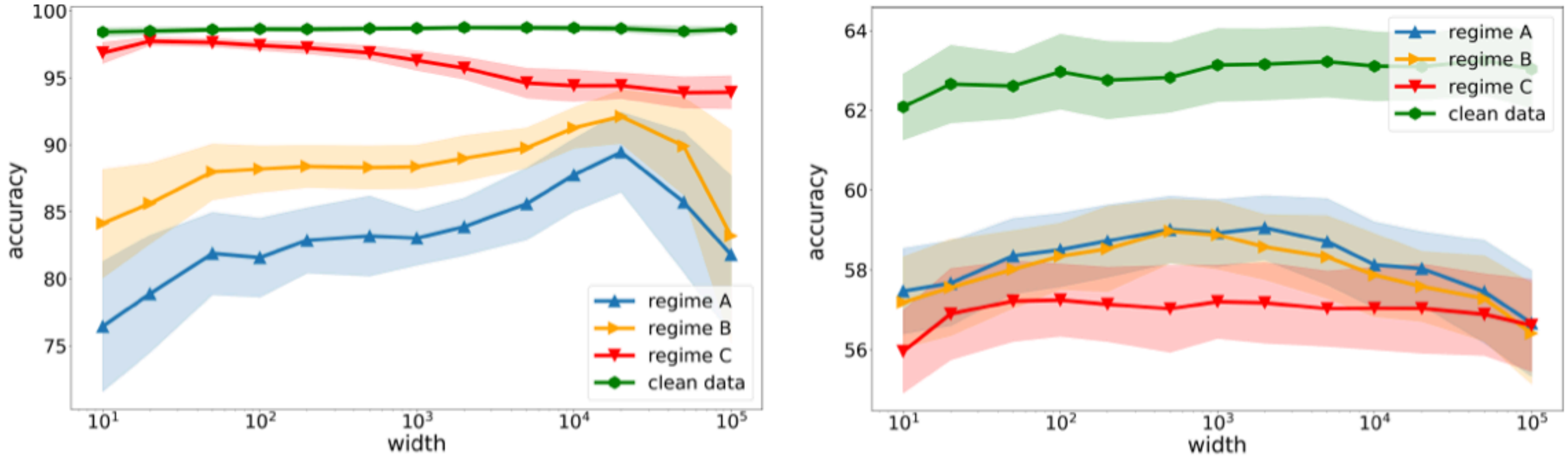


Figure: Clean test accuracy as a function of network width under clean data setting and poisoned data setting on MNIST (left) and CIFAR10 (right).

Reference

1. Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).

2. Gu, Tianyu, Brendan Dolan-Gavitt, and Siddharth Garg. "Badnets: Identifying vulnerabilities in the machine learning model supply chain." *arXiv preprint arXiv:1708.06733* (2017).

3. Koh, Pang Wei, and Percy Liang. "Understanding black-box predictions via influence functions." *International Conference on Machine Learning*. PMLR, 2017.

4. Chan, Patrick PK, et al. "Causative label flip attack detection with data complexity measures." *International Journal of Machine Learning and Cybernetics* 12.1 (2021): 103-116.

5. Du, Simon S., et al. "Gradient descent provably optimizes over-parameterized neural networks." *arXiv preprint arXiv:1810.02054* (2018).

6. Ji, Ziwei, and Matus Telgarsky. "Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow relu networks." *arXiv preprint arXiv:1909.12292* (2019).