上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

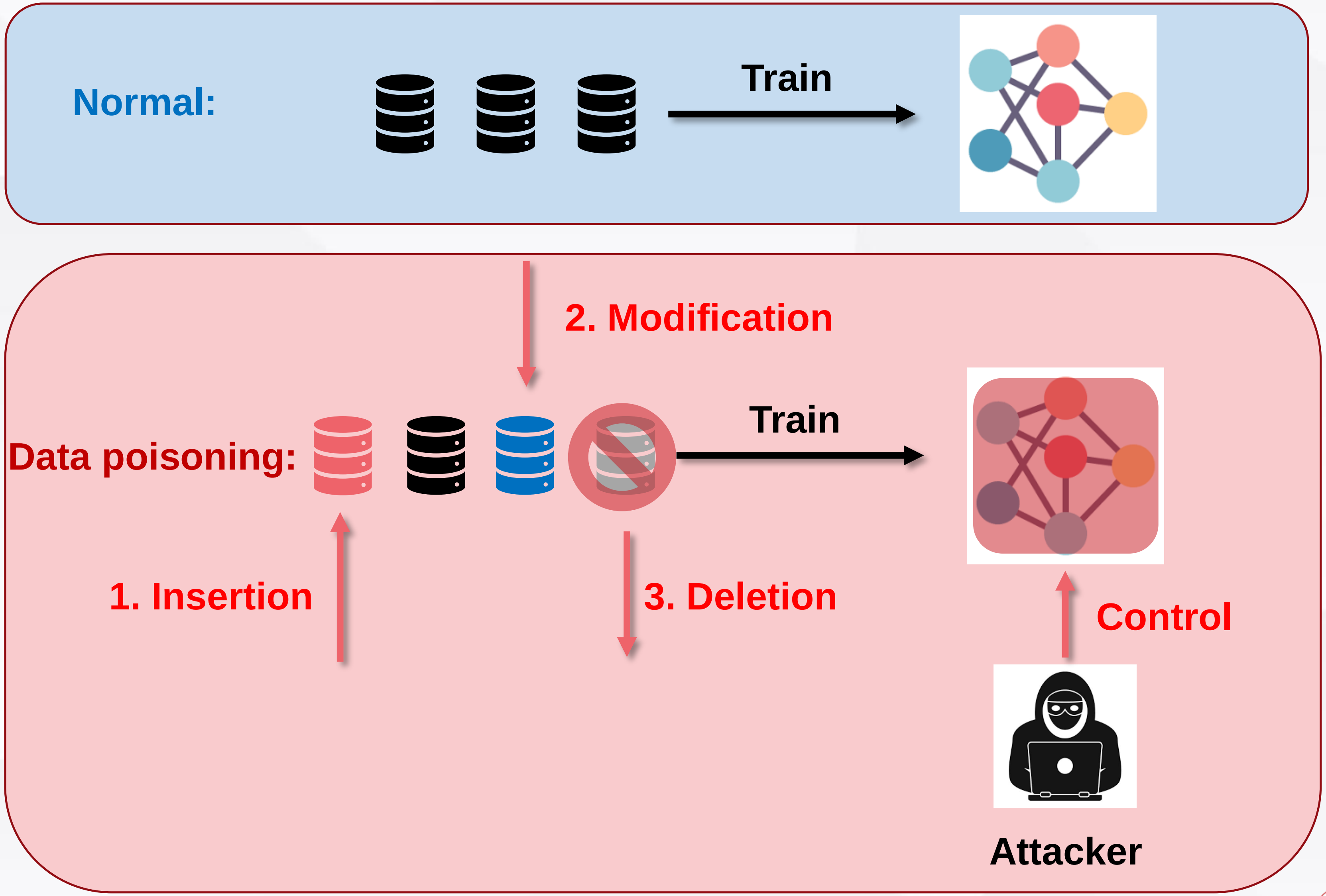
On Collective Robustness of Bagging Against Data Poisoning

Ruoxin Chen, Zenan Li, Jie Li, Chentao Wu, Junchi Yan

Shanghai Jiao Tong University

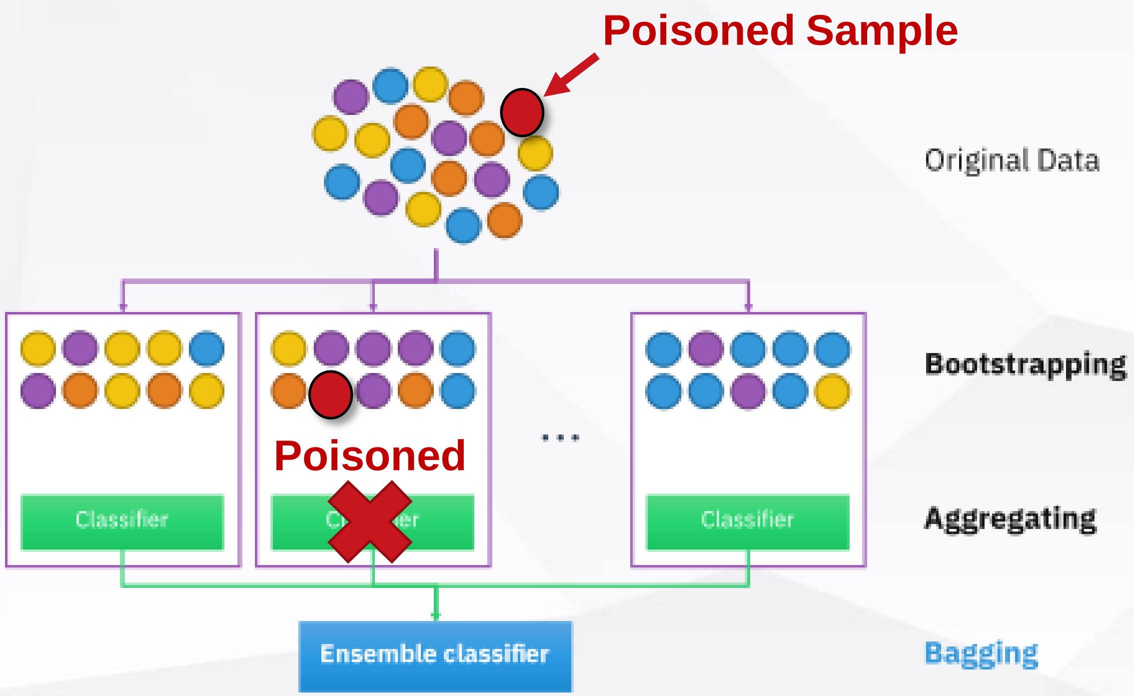
饮 水 思 源 • 爱 国 荣 校

Data Poisoning Attacks



The Only Certified Defense: Bagging

- Bagging is **the only model-agnostic certified defense** against sample-level data poisoning attacks. In fact, all three model-agnostic certified defenses (Levine & Feizi, 2021; Jia et al., 2021; Wang et al., 2022) are the specific variants of bagging.



From https://en.wikipedia.org/wiki/Bootstrap_aggregating

Certified robustness of bagging is from:

- Mechanism 1: a poisoned sample can only influence a bounded number of sub-classifiers (**the influence range of data poisoning is limited**) .
- Mechanism 2: the existing gap between the top1 votes and the "runner-up" votes can tolerate a bounded number of vote manipulation (**the intrinsic robustness from the voting mechanism**) .

Our Contributions

- We propose the first **collective certification** for bagging, to certify its collective robustness against data poisoning.

- We propose **hash bagging** to improve the collective robustness for bagging.

Sample-wise Robustness V.S. Collective Robustness

- **Threat model (sample-wise robustness):** the attacker has full knowledge about the trainset, the testing sample (denoted by x_0), the training details, and the model architecture.

- **Threat model (collective robustness):** the attacker has full knowledge about the trainset, the M-size testset (denoted by D_{test}), the training details, and the model architecture.

- **Poison budget:** the attacker can arbitrarily insert r_{ins}, delete r_{del} and modify r_{mod} samples
- **Certified sample-wise robustness:** guarantee that the prediction on x_0 is unchangeable to any poisoning attack subject to the poison budget constraint.
- **Certified collective robustness:** guarantee the minimum number of unchanged predictions.

Why Need Collective Robustness ?

- Fundamental difference: the setting of the attacker objective
  - 1) **sample-wise robustness** assumes the attacker aims to change the single prediction.
  - 2) **collective robustness** assumes the attacker aims to degrade the overall accuracy on the testset.

- **i) Collective Robustness Is More Practical:** most data poisoning works [Wang & Chaudhuri, 2018; Goldblum et al., 2022; Geiping et al., 2020; Huang et al., 2020; Shafahi et al., 2018; Wang et al., 2022] focus on degrading the overall testing accuracy, which exactly corresponds to collective robustness.

- **ii) Collective Robustness Is More General:** sample-wise robustness is a special case of collective robustness when the testset size is one.

- **iii) Collective Robustness Is More Stable:** the collective robustness on two similar testsets is close while sample-wise robustness is different from sample to sample greatly.

Collective Robustness Certification for Bagging

Proposition 1 (Certified collective robustness of vanilla bagging). *For testset $\mathcal{D}_{test} = \{x_j\}_{j=0}^{M-1}$, we denote $\hat{y}_j = g(x_j)$ $(j = 0, \ldots, M-1)$ the original ensemble prediction, and $\mathcal{S}_i = \{g \mid s_i \in \mathcal{D}_g\}$ the set of the indices of the sub-trainsets that contain s_i (the i-th training sample). Then, the maximum number of simultaneously changed predictions (denoted by M_{ATK}) under r_{mod} adversarial modifications, is computed by* $(\mathbf{P1})$:

$$(\mathbf{P1}): \quad M_{\text{ATK}} = \max_{P_0,\ldots,P_{N-1}} \sum_{x_j \in \mathcal{D}_{test}} \mathbb{I}\Big\{\overline{V}_{x_j}(\hat{y}_j) < \max_{y \neq \hat{y}_j} \Big[\overline{V}_{x_j}(y) + \frac{1}{2}\mathbb{I}\{y < \hat{y}_j\}\Big]\Big\} \tag{2}$$

$$\text{s.t.} \quad [P_0, P_1, \ldots, P_{N-1}] \in \{0,1\}^N \tag{3}$$

$$\sum_{i=0}^{N-1} P_i \leq r_{\text{mod}} \tag{4}$$

$$\overline{V}_{x_j}(\hat{y}_j) = \underbrace{V_{x_j}(\hat{y}_j)}_{\textit{Original votes}} - \underbrace{\sum_{g=0}^{G-1} \mathbb{I}\{g \in \bigcup_{\forall i, P_i = 1} \mathcal{S}_i\}\mathbb{I}\{f_g(x_j) = \hat{y}_j\}}_{\textit{Influenced votes}}$$
$$\forall x_j \in \mathcal{D}_{test}, \; \hat{y}_j = g(x_j) \tag{5}$$

$$\overline{V}_{x_j}(y) = \underbrace{V_{x_j}(y)}_{\textit{Original votes}} + \underbrace{\sum_{g=0}^{G-1} \mathbb{I}\{g \in \bigcup_{\forall i, P_i = 1} \mathcal{S}_i\}\mathbb{I}\{f_g(x_j) \neq y\}}_{\textit{Influenced votes}}$$
$$\forall x_j \in \mathcal{D}_{test}, \; \forall y \in \mathcal{Y}, y \neq \hat{y}_j \tag{6}$$

The certified collective robustness is $M - M_{\text{ATK}}$.

Eq. (2): the objective is to maximize the number of simultaneously changed predictions. Note that a prediction is changed if there exists another class with more votes.

Eq. (3): $[P_0, \ldots, P_{N-1}]$ are the binary variables that represent the poisoning attack, where $P_i = 1$ means that the attacker poisons the training sample s_i among the trainset $D_{train} = \{s_i\}_{i=0}^{N-1}$.

Eq. (4): the number of modifications is bounded within r_{mod}.

Eq. (5): $\overline{V}_{x_i}(y_j)$ denotes the minimum number of votes for class $\hat{y}_j$ (after being attacked), equals to the original value minus the number of the influenced sub-classifiers whose original predictions are $\hat{y}_j$.

Eq. (6): $\overline{V}_{x_i}(y), y \neq y_i$, the maximum number of votes for class y: $y \neq y_j$ (after being attacked), equals to the original value plus the number of influenced sub-classifiers whose original predictions are not y, because that, under our threat model, the attacker is allowed to arbitrarily manipulate the predictions of those influenced sub-classifiers.

SJTU

Upper Bound of Tolerable Poison Budget

Proposition 2 (Upper bound of tolerable poison budget). *Given* $\mathcal{S}_i = \{g \mid s_i \in \mathcal{D}_g\}$ *(*$i = 0, \ldots, N-1$*)*, the upper bound of the tolerable poisoned samples *(denoted by* $\bar{r}$*) is*

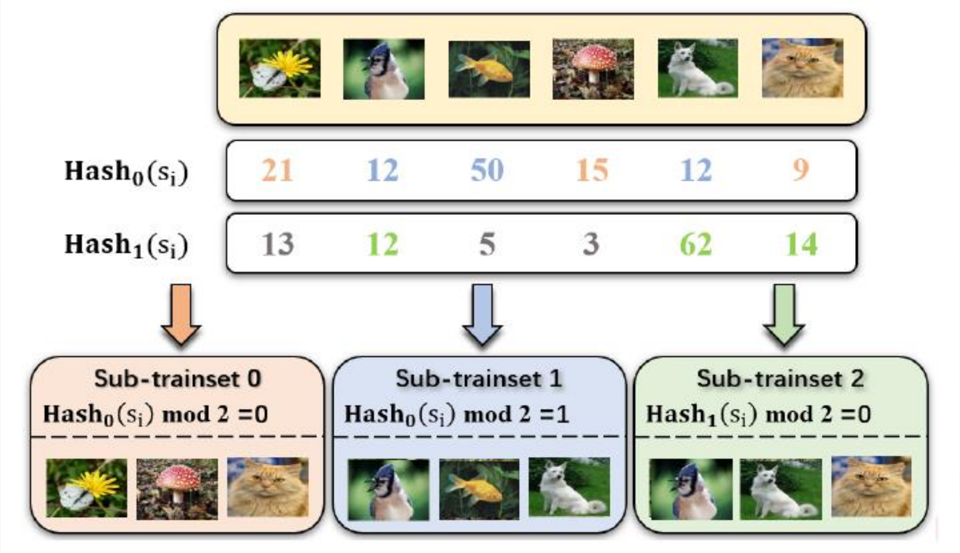
$$\bar{r} = \min |\Pi| \ s.t. \ |\bigcup_{i \in \Pi} \mathcal{S}_i| > G/2 \tag{7}$$

where Π *denotes a set of indices. The upper bound of the tolerable poisoned samples equals the minimum number of training samples that can influence more than a half of sub-classifiers.*

- Proposition 2 states that the tolerable poison budget is no larger than $\bar{r}$.
- We enlarge $\bar{r}$ to improve collective robustness.
- A way of enlarging $\bar{r}$ is to bound the influence scope for each training sample. In particular, if each training sample is only contained in Γ sub-trainsets (bound the influence scope), we can guarantee $\bar{r} \geq N/(2\Gamma)$.
- Therefore, we design a form of bagging, improving (both collective and sample-wise) certified robustness by constraining the influence scope for each training sample

Hash Bagging Improves Collective Robustness

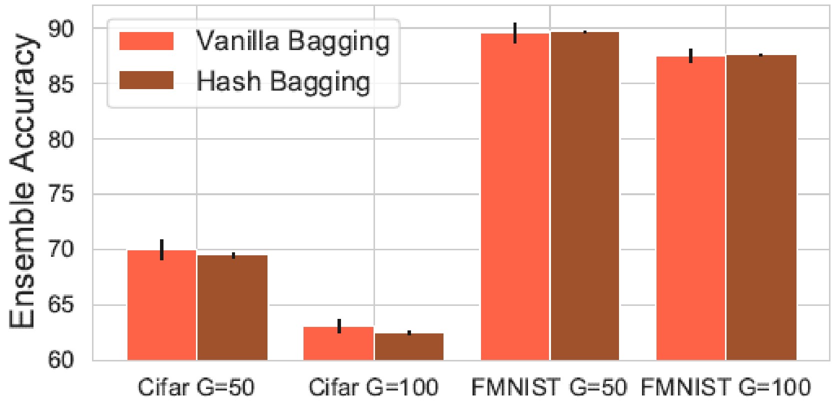
Hash Bagging



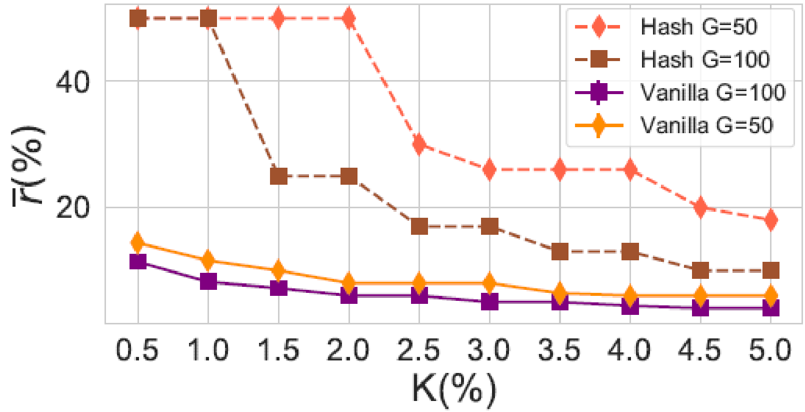
- Hash bagging when $N = 6$ (trainset size), $K = 3$ (sub-trainset size), $G = 3$ (number of sub-trainsets).
  - 0-th sub-trainset: $Hash_0(s_i)\ mod\ 2 = 0$ (the samples whose hash values are colored by red).
  - 1-st sub-trainset: $Hash_0(s_i)\ mod\ 2 = 0$ (the samples whose hash values are colored by blue).
  - 2-nd sub-trainset: $Hash_1(s_i)\ mod\ 2 = 0$ (the samples whose hash values are colored by green).

Hash bagging is one of the bagging forms with the smallest poisoning influence scope.

Experiments: Hash Bagging V.S. Vanilla Bagging



(a) Comparison on ensemble accuracy ($K = N/G$).



(b) Comparison on $\overline{r}$ on FMNIST.

- 1. Hash bagging achieves a **comparable** ensemble accuracy.
- 2. Hash bagging achieves a **much larger** tolerable poison budget.

Experiments: Collective Certification V.S. Sample-wise Certification

G	Bagging	Certification	Metric	5%	10%	15%	20%	25%
20	Vanilla	Sample-wise	CR	9230	0	0	0	0
			CA	7321	0	0	0	0
		Collective	CR	9348	0	0	0	0
			M_{ATK}	↓15.3%	NaN	NaN	NaN	NaN
			CA	7394	0	0	0	0
			M_{ATK}	↓17.5%	NaN	NaN	NaN	NaN
	Hash	Sample-wise	CR	9858	9738	9602	9461	9293
			CA	7681	7621	7538	7462	7362
		Collective	CR	**9915**	**9821**	**9726**	**9608**	**9402**
			M_{ATK}	↓40.1%	↓31.7%	↓31.1%	↓27.3%	↓23.9%
			CA	**7701**	**7663**	**7608**	**7547**	**7458**
			M_{ATK}	↓34.5%	↓35.6%	↓34.8%	↓30.7%	↓25.5%
40	Vanilla	Sample-wise	CR	9482	8648	0	0	0
			CA	7466	6986	0	0	0
		Collective	CR	9566	8817	0	0	0
			M_{ATK}	↓16.2%	↓12.5%	NaN	NaN	NaN
			CA	7513	7086	0	0	0
			M_{ATK}	↓16.5%	↓13.1%	NaN	NaN	NaN
	Hash	Sample-wise	CR	9873	9769	9636	9491	9366
			CA	7681	7625	7546	7459	7399
		Collective	CR	**9919**	**9842**	**9755**	**9601**	**9461**
			M_{ATK}	↓36.2%	↓31.6%	↓32.7%	↓21.6%	↓15.0%
			CA	**7700**	**7661**	**7613**	**7536**	**7457**
			M_{ATK}	↓27.5%	↓28.8%	↓32.8%	↓26.5%	↓16.5%

Electricity Dataset

G	Bagging	Certification	Metric	5%	10%	15%	20%	25%
50	Vanilla	Sample-wise	CR	7432	0	0	0	0
			CA	7283	0	0	0	0
		Collective	CR	7727	0	0	0	0
			M_{ATK}	↓11.5%	NaN	NaN	NaN	NaN
			CA	7515	0	0	0	0
			M_{ATK}	↓13.8%	NaN	NaN	NaN	NaN
	Hash	Sample-wise	CR	9576	9307	8932	8671	8238
			CA	8768	8635	8408	8246	7943
		Collective	CR	**9726**	9410	9024	8761	8329
			M_{ATK}	↓35.4%	↓14.9%	↓8.61%	↓6.77%	↓5.16%
			CA	**8833**	**8719**	8493	8327	8022
			M_{ATK}	↓32.8%	↓25.4%	↓15.2%	↓11.2%	↓7.72%
		Decomposition	CR	9666	**9472**	**9124**	**8887**	**8491**
			M_{ATK}	↓21.2%	↓23.8%	↓18.0%	↓16.2%	↓14.4%
			CA	8812	8716	**8527**	**8385**	**8119**
			M_{ATK}	↓22.2%	↓24.5%	↓21.3%	↓19.3%	↓17.2%
100	Vanilla	Sample-wise	CR	7548	0	0	0	0
			CA	7321	0	0	0	0
		Collective	CR	8053	0	0	0	0
			M_{ATK}	↓20.6%	NaN	NaN	NaN	NaN
			CA	7746	0	0	0	0
			M_{ATK}	↓29.4%	NaN	NaN	NaN	NaN
	Hash	Sample-wise	CR	9538	9080	8653	8249	7823
			CA	8554	8316	8049	7797	7486
		Collective	CR	9611	9167	8754	8344	7912
			M_{ATK}	↓15.8%	↓9.46%	↓7.50%	↓5.42%	↓4.09%
			CA	**8610**	8375	8116	7857	7558
			M_{ATK}	↓26.7%	↓13.2%	↓9.37%	↓6.20%	↓5.63%
		Decomposition	CR	**9631**	**9232**	**8837**	**8450**	**8036**
			M_{ATK}	↓20.1%	↓16.5%	↓13.6%	↓11.5%	↓9.78%
			CA	8595	**8407**	**8152**	**7917**	**7639**
			M_{ATK}	↓19.5%	↓20.3%	↓14.4%	↓12.4%	↓12.0%

FMNIST Dataset

G	Bagging	Certification	Metric	5%	10%	15%	20%	25%
50	Vanilla	Sample-wise	CR	2737	0	0	0	0
			CA	2621	0	0	0	0
		Collective	CR	3621	0	0	0	0
			M_{ATK}	↓12.2%	NaN	NaN	NaN	NaN
			CA	3335	0	0	0	0
			M_{ATK}	↓16.3%	NaN	NaN	NaN	NaN
	Hash	Sample-wise	CR	8221	7268	6067	5320	4229
			CA	6305	5864	5186	4705	3884
		Collective	CR	8393	7428	6204	5435	4290
			M_{ATK}	↓9.67%	↓5.86%	↓3.48%	↓2.46%	↓1.06%
			CA	6410	5985	5342	4848	4006
			M_{ATK}	↓15.2%	↓10.7%	↓8.62%	↓6.24%	↓3.92%
		Decomposition	CR	**8694**	**7854**	**6686**	**5912**	**4826**
			M_{ATK}	↓26.6%	↓21.4%	↓15.7%	↓12.6%	↓10.3%
			CA	**6490**	**6147**	**5553**	**5113**	**4341**
			M_{ATK}	↓26.8%	↓25.0%	↓20.2%	↓17.8%	↓14.7%
100	Vanilla	Sample-wise	CR	2621	0	0	0	0
			CA	1876	0	0	0	0
		Collective	CR	2657	0	0	0	0
			M_{ATK}	↓7.93%	NaN	NaN	NaN	NaN
			CA	2394	0	0	0	0
			M_{ATK}	↓11.8%	NaN	NaN	NaN	NaN
	Hash	Sample-wise	CR	7685	5962	4612	3504	2593
			CA	5396	4571	3787	3008	2315
		Collective	CR	7744	5974	4618	3509	2598
			M_{ATK}	↓2.54%	↓0.30%	↓0.11%	↓0.08%	↓0.07%
			CA	5475	4650	3825	3030	2330
			M_{ATK}	↓9.21%	↓4.69%	↓1.54%	↓0.68%	↓0.38%
		Decomposition	CR	**8137**	**6469**	**5061**	**4035**	**2987**
			M_{ATK}	↓19.5%	↓12.5%	↓8.33%	↓8.17%	↓5.32%
			CA	**5570**	**4841**	**4098**	**3338**	**2635**
			M_{ATK}	↓20.3%	↓16.0%	↓12.6%	↓10.2%	↓8.12%

CIFAR-10 Dataset

Collective certification consistently certifies a much tighter M_{ATK} (the maximum number of simultaneously changed predictions) than the sample-wise certification

Thanks

Paper: https://arxiv.org/abs/2205.13176

Github: https://github.com/Emiyalzn/ICML22-CRB

Contact: Ruoxin Chen, chenruoxin@sjtu.edu.cn

Jie Li, lijiecs@sjtu.edu.cn

饮 水 思 源 爱 国 荣 校