Poison is not Traceless: Black-Box Detection of Label Poisoning Attacks

anonymous

I. Supplementary Material

To ensure reproducibility, we provide additional details for the paper in this section.

A. Full List of Complexity Measures

The full set of measures is listed in Table I We also include the *Standard Deviations* (SDs) when possible, which are not listed in the table. These measures include F1, N2, N3, N4, T1, and Hubs. As a result, there are 28 measures in total.

TABLE I
List of measures in C-Measures. If possible, Standard Deviations (SDs) of measures are also included, but are not listed.

Category	Acronym	Description
Feature-based	F1	Maximum Fisher's discriminant ratio
	F1v	Directional-vector maximum Fisher's discriminant ratio
	F2	Volume of overlapping region
	F3	Maximum individual feature efficiency
	F4	Collective feature efficiency
Linearity	L1	Sum of the error distance by linear programming
	L2	Error rate of the linear SVM classifier
	L3	Non-linearity of the linear SVM classifier
Neighborhood	N1	Fraction of borderline points
	N2	Ratio of intra/extra class nearest-neighbors distance
	N3	Error rate of nearest-neighbors classifier
	N4	Non-linearity of nearest-neighbors classifier
	T1	Fraction of hyperspheres covering data
	LSC	Local Set average Cardinality
Network	Density	Average density of the network
	ClsCoef	Clustering Coefficient
	Hubs	Hub score – Number of connections each node has
Dimensionality	T2	Average number of features per dimension
	T3	Average number of PCA dimensions per points
	T4	Ratio of the PCA dimension to the original dimension
Class Imbalance	C1	Entropy of classes proportions
	C2	Imbalance ratio

B. Details of FALFA

We obtain the multiplier λ by generalizing all the combinations between y_i and y'_i in a binary classification task, as shown in Table II.

TABLE II
All combinations for $|y'_i - y_i|$. By introducing a multiplier λ, $\lambda \cdot (y'_i - y_i)$ is equivalent to $|y'_i - y_i|$.

y_i	y'_i	$\|y'_i - y_i\|$	λ
0	0	0	1
0	1	1	1
1	0	-1	-1
1	1	0	-1

C. Time Complexity of FALFA

FALFA is more computationally efficient than ALFA [5], [7] and PoisSVM by a substantial margin. Linear programming is an exponential-time algorithm, the time complexity is around $O(n^{2.5})$ [6]. Xiao *et al.*'s ALFA creates a copy of $\mathcal{Y}_{\text{tr}}$ in the linear programming step, so n is essentially doubled. Paudice *et al.*'s ALFA on NN is slower than Xiao *et al.*'s, since it traverses all combinations of $\mathcal{Y}_{\text{tr}}$ instead of using linear programming. FALFA uses linear programming but without doubling $\mathcal{Y}_{\text{tr}}$, resulting in an approximately $2^{2.5} \approx 5.6$ times faster than ALFA on each iteration. Our test shows that FALFA converges at 2 iterations on average, but ALFA takes more than 5 iterations to converge. In the worst-case scenario, FALFA completes CMC dataset at 22.4 ± 8.6 secs, while ALFA completes the same dataset at 405.8 ± 348.4 secs, and PoisSVM took over 2 hours to compute the same dataset. We observe the minimal difference on Breastcancer, where FALFA completes the task at $5.3{\pm}1.9$ secs, and it takes ALFA $7.4{\pm}5.6$ secs.

D. Hardware and Software Configurations

All experiments are conducted on a workstation with the following configurations:

- CPU: AMD Ryzen 9 5900 24 threads @ 4.4GHz
- GPU: Nvidia GeForce RTX 3090 24GB
- Memory: 64GB
- Operating System: Ubuntu 20.04.3 LTS
- Software: Python 3.8.10, PyTorch 1.10.1+cu113, scikit-learn 1.0.2

The baseline data poisoning attacks are obtained from *Adversarial Robustness Toolbox* (ART) 1.9.1 [4] and *Secure and Explainable Machine Learning in Python* (SecML) 0.15 [3].

The execution time mentioned in the paper is evaluated using the environment above.

E. Datasets

All datasets are obtained from the UCI Machine Learning Repository [2]. We apply standardization on all datasets during the preprocessing.

For multi-class classification tasks, we convert the dataset into binary based on the following:

- **Abalone:** If the 'Rings' attribute is less than 10, we assign the example to the negative class; Else, assign to the positive class. We exclude the categorical attribute – 'Sex' and the output label – 'Rings' from the inputs.
- **CMC:** has 3 output classes: 1) No-use, 2) Long-term, and 3) Short-term. If the class is 'No-use', assign it to the negative class; Else, to the positive class.
- **Texture:** It has 10 output classes. We use a subset which contains examples labeled as '3' and '9'. If the class is '3', assign it to the negative class; Else, to the positive class.
- **Yeast:** It has 10 output classes. We select ''0 and '7', the top two classes sorted by sample size. If the class is '0', assign it to the negative class; Else, to the positive class.

F. Additional Results

Performance Loss. Fig. 1 shows the performance loss on all real datasets.

Performance Loss at a Low Poisoning Rate. Here, we present the performance loss at a low poisoning rate (10%) in Table III. This is the test accuracy difference before and after the attack. PoisSVM has no meaningful impact ($< 2\%$) on the classifiers' performance in 7 out of 10 datasets. Meanwhile, SLN leads to minor performance improvement on CMC and Yeast. This result matches Chen *et al.*'s prior work [1], which shows DNNs are resilient to a low amount of label noise.

TABLE III
PERFORMANCE LOSS (%) AFTER ATTACKED BY A POISONING ATTACK WITH 10% POISONING RATE.

Dataset	SLN	PoisSVM	ALFA	FALFA
Abalone	0.8 ± 0.7	1.8 ± 0.8	9.5 ± 1.9	7.7 ± 1.7
Australian	0.7 ± 0.5	4.5 ± 3.9	4.9 ± 4.0	8.3 ± 3.8
Banknote	1.4 ± 2.3	1.1 ± 1.1	10.9 ± 2.5	10.3 ± 2.9
Breastcancer	2.5 ± 0.7	5.3 ± 4.6	7.2 ± 2.0	9.1 ± 2.7
CMC	-0.2 ± 0.7	15.1 ± 4.7	3.5 ± 3.0	5.7 ± 3.3
HTRU2	0.7 ± 0.3	0.7 ± 1.3	9.2 ± 3.1	9.4 ± 2.4
Phoneme	3.5 ± 2.9	0.9 ± 2.1	6.8 ± 0.7	11.6 ± 2.1
Ringnorm	0.1 ± 0.3	1.7 ± 0.5	3.2 ± 2.5	6.4 ± 2.9
Texture	0.5 ± 1.1	1.2 ± 0.8	7.9 ± 4.6	4.9 ± 3.9
Yeast	-0.2 ± 1.6	1.9 ± 3.8	10.4 ± 4.9	2.3 ± 4.6

REFERENCES

[1] P. Chen, G. Chen, J. Ye, P.-A. Heng *et al.*, "Noise against noise: stochastic label noise helps combat inherent label noise," in *International Conference on Learning Representations*, 2020.

[2] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[3] M. Melis, A. Demontis, M. Pintor, A. Sotgiu, and B. Biggio, "secml: A python library for secure and explainable machine learning," *arXiv preprint arXiv:1912.10013*, 2019.

[4] M.-I. Nicolae, M. Sinn, M. N. Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. Molloy, and B. Edwards, "Adversarial robustness toolbox v1.2.0," *CoRR*, vol. 1807.01069, 2018. [Online]. Available: https://arxiv.org/pdf/1807.01069

[5] A. Paudice, L. Muñoz-González, and E. C. Lupu, "Label sanitization against label flipping poisoning attacks," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 5–15.

[6] P. M. Vaidya, "Speeding-up linear programming using fast matrix multiplication," in *30th annual symposium on foundations of computer science*. IEEE Computer Society, 1989, pp. 332–337.

[7] H. Xiao, H. Xiao, and C. Eckert, "Adversarial label flips attack on support vector machines," in *ECAI 2012*. IOS Press, 2012, pp. 870–875.

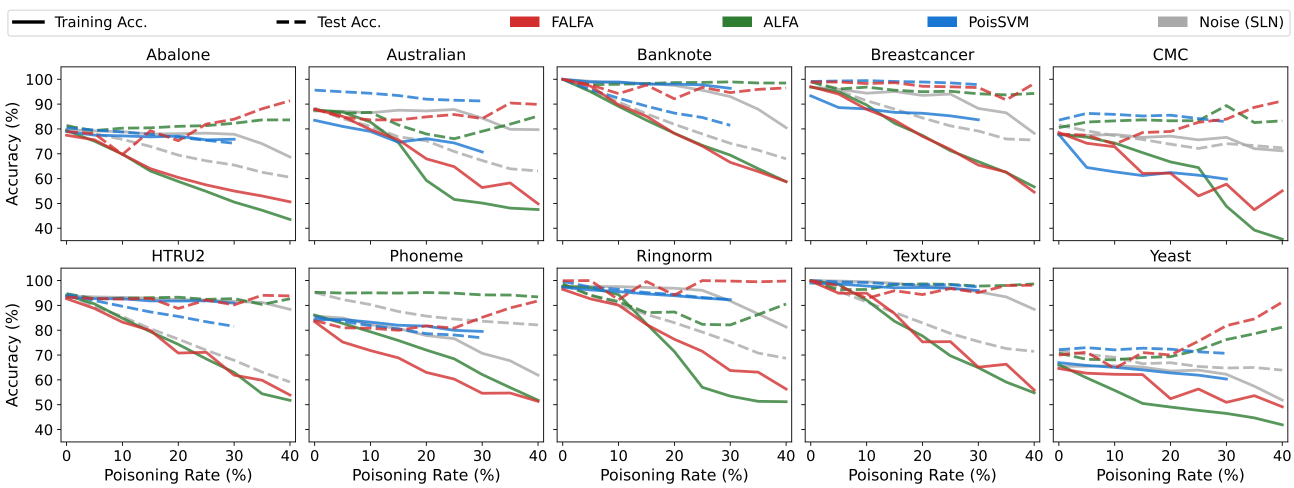

Fig. 1. Train and test accuracy at various poisoning rates when classifiers under SLN, PoisSVM, and FALFA attacks.