In Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), Venice, Italy.

Infinity-norm Support Vector Machines against Adversarial Label Contamination

Ambra Demontis, Battista Biggio,
Giorgio Fumera, Giorgio Giacinto, and Fabio Roli

Dept. of Electrical and Electronic Eng. University of Cagliari, Piazza d'Armi, 09123, Cagliari, Italy
ambra.demontis, battista.biggio, fumera, giacinto, roli @diee.unica.it

Abstract

Nowadays machine-learning algorithms are increasingly being applied in security-related applications like spam and malware detection, aiming to detect never-before-seen attacks and novel threats. However, such techniques may expose specific vulnerabilities that may be exploited by carefully-crafted attacks. Support Vector Machines (SVMs) are a well-known and widely-used learning algorithm. They make their decisions based on a subset of the training samples, known as support vectors. We first show that this behaviour poses risks to system security, if the labels of a subset of the training samples can be manipulated by an intelligent and adaptive attacker. We then propose a countermeasure that can be applied to mitigate this issue, based on infinity-norm regularization. The underlying rationale is to increase the number of support vectors and balance more equally their contribution to the decision function, to decrease the impact of the contaminating samples during training. Finally, we empirically show that the proposed defence strategy, referred to as Infinity-norm SVM, can significantly improve classifier security under malicious label contamination in a real-world classification task involving malware detection.

1 Introduction

Machine learning techniques are increasingly being used in different application fields, including medicine, economy, and computer security. In security-related applications, like malware detection or biometric authentication, a system may be targeted by an intelligent attacker who aims at evading detection of "malicious" samples (*e.g.*, a file containing malware, or the attempt by an unauthorized user to access a resource protected by a biometric system), exploiting knowledge of the underlying learning algorithm. This issue is addressed by the novel research field of *adversarial machine learning*, whose aim is to investigate the vulnerabilities of machine learning algorithms, and to improve their security in adversarial environments (see, *e.g.*, [1, 11, 4, 3, 2]).

Attacks against machine learning algorithms can be categorized into *evasion* and *poisoning*. Evasion attacks consist of modifying malicious samples at test time to avoid detection. Poisoning attacks are performed at training time, instead, when the classifier learns from a set of labelled samples how to perform the desired classification task (i.e., discriminating between malicious and legitimate samples): they consist in creating well-crafted samples and injecting them into the training data to subvert the system function. Depending on the attacker's capability, poisoning attacks can be carried out by manipulating either a sample (thus changing its feature values) or its label. In the latter case the attack is named *adversarial label flip* [12, 20]. The capability of changing the labels of the training samples is potentially very harmful, as they directly impact classifier learning [9].

Label-flip attacks are of practical relevance, as the attacker may have access to the training labels in a wide range of applications in which systems ask users to provide a feedback on the classified samples for improving their recognition capability. For instance, server-side spam filters allow users to correct the label (spam or legitimate) automatically assigned to incoming emails, if wrong; an attacker may exploit this feature by creating an email account on a provider protected by the targeted spam filter, and then purposely mislabeling incoming emails that will be subsequently used to retrain the classifier, to gradually poison classifier training. Another instance is PDFRate,[1] which is an online tool for detecting malware embedded into PDF files [16]: an attacker may provide wrong feedback to the system, which amounts to manipulating the labels of (future) training samples. Since collecting labels from domain experts is usually costly, crowdsourcing systems like Amazon Mechanical Turk are being used to assign this task to non-expert individuals: this scenario may be exploited by attackers to provide wrong labels. "Malicious crowdsourcing" or "crowdturfing" services are growing in popularity: Internet users are payed to perform profitable malicious tasks, like spam dissemination, including polluting the data used as training samples by machine learning systems [19].

The above examples show that understanding label-flip attacks more thoroughly and finding effective countermeasures is a very relevant research topic. In this paper we focus on label-flip attacks against Support Vector Machines (SVM), which are a state-of-the-art, widely used classifier. Previous work, summarized in Sect. 2.2, has shown that SVM classification accuracy decreases in the presence of label noise (even non-adversarial), and that some SVM variants are more robust under random label flips. To our knowledge the only specific countermeasure against label-flip attacks has been proposed in our previous work [5]: it is a heuristic approach that enforces the classifier to evenly weigh all the training samples, to increase the stability of the decision function with respect to changes of the training labels. In this work we give a theoretical support to the above approach, and propose a general, more theoretically-sound countermeasure (Sect. 3) rooted in recent findings about the relationship between regularized and robust optimization (Sect. 2.3), which also reduces the complexity of SVM training. In Sect. 4 we validate our approach on artificial and real-world data sets In Sect. 5 we review related work, and in Sect. 6 we discuss the main contribution of this work and some interesting research directions.

2 Background

We first describe two label flip strategies we shall consider in our experiments. Then we review state-of-the-art strategies that may be used to improve SVM security under label flip attacks. We finally overview works highlighting a link between regularization and robustness, that will provide a formal support to the approach we propose in Sect. 3.

In the following we denote by $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{\mathsf{n}}$ the training set, where $\boldsymbol{x}_i \in \mathbb{R}^{\mathsf{d}}$ is the feature vector of the i-th sample and $y_i \in \{-1, +1\}$ its label (respectively, for legitimate and malicious samples). The decision function of a trained SVM classifier (using a nonlinear kernel) is $g(\boldsymbol{x}) = \sum_{i=1}^{\mathsf{n}} \alpha_i y_i k(\boldsymbol{x}, \boldsymbol{x}_i) + b$, where $k(\cdot, \cdot)$ is the kernel function, and $\{\alpha_i\}_{i=1}^{\mathsf{n}}$ and b are coefficients set by the learning algorithm.

2.1 Label-Flip Attacks

We consider two different kinds of label flip attacks. In both cases we set a constraint to the fraction of labels the adversary can change, to reflects a likely limitation in real-world scenarios.

Random label flip is a baseline attack, which consists of flipping the labels of a randomly-chosen fraction of training samples, without exploiting any knowledge of the targeted classifier.

[1] Available at: http://pdfrate.com

Adversarial Label-Flip Attack (ALFA-Tilt) is a different attack proposed in [5, 21], and assumes a skilled attacker whose aim is to maximize the classifier error on untainted (testing) data. Since finding the subset of samples whose label flipping maximizes the testing error is a non-trivial problem, the authors devised a heuristic approach that maximizes a surrogate measure of the testing error, namely, the angle between the decision hyperplane found by the untainted classifier and the one under attack.

Different attack scenarios can be simulated, depending on the attacker's level of knowledge of the system: either perfect knowledge, if the attacker exactly knows the coefficients of the SVM decision function, or limited knowledge, if she is only capable of creating a data set sampled from the same distribution of the one used for training the original classifier, and then training a surrogate classifier for estimating the original decision function. In this work we consider the worst case of perfect knowledge, although in real scenarios the attacker is likely to have only a limited knowledge of the system.

2.2 SVM Variants

A possible countermeasure to label-flip attacks is to enforce the decision function of an SVM to weigh more uniformly the contribution of each training sample to the decision hyperplane. The reason is that this would decrease the impact of each single point during learning of the decision function. Two SVM variants can be used to this aim.

Least-Squares SVM (LS-SVM). This SVM variant [18] uses a quadratic loss function instead of the hinge loss. This makes its solution non-sparse, *i.e.*, all the training samples are assigned a non-null α value. In particular, the LS-SVM (primal) learning problem is:

$$\min_{\boldsymbol{w},b,\boldsymbol{e}} \ \tfrac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} + \gamma \tfrac{1}{2}\textstyle\sum_{i=1}^{\mathsf{n}} e_i^2 \quad s.t.\ y_i = \boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b + e_i \ \forall i\,, \tag{1}$$

where ϕ is the kernel-induced feature mapping, and $\boldsymbol{w}$ the set of primal weights. Recall that, as in SVM learning, $\boldsymbol{w} = \sum_{i=1}^{\mathsf{n}} \alpha_i y_i \phi(\boldsymbol{x}_i)$ and $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}_j)$, which enables learning of nonlinear decision functions in input space by solving the corresponding dual optimization problem (*i.e.*, optimizing directly the dual variables α instead of $\boldsymbol{w}$).

Label Noise Robust SVM (LN-robust SVM). This is another SVM variant proposed in [5] against label flip attacks. It assumes that the label of each training sample can be independently flipped with the same probability μ. The probability of label flips is then encoded into the kernel matrix, which is involved in the dual SVM learning problem. The expected value of the modified kernel matrix (which is still positive-semidefinite) is then used for solving the standard SVM learning problem. It turns out that, by increasing the variance $S = \mu(1-\mu)$, the variance of the coefficients α_i decreases; accordingly, each training sample is more likely to become a support vector, providing a more balanced contribution to the decision function. This approach only requires a simple correction to the kernel matrix with respect to standard SVM. It is however a heuristic solution, which also requires one to be able to reliably estimate the fraction of potential label flips in the training data.

2.3 Robustness and Regularization

Recently an interesting relationship between regularized and robust optimization problems has been pointed out [22]. Under mild assumptions, the two kind of problems are equivalent. In particular, the robust optimization problem considered in [22] is:

$$\min_{\boldsymbol{w},b} \ \max_{\boldsymbol{u}_1,..,\boldsymbol{u}_{\mathsf{n}} \in \mathcal{U}} \textstyle\sum_{i=1}^{\mathsf{n}} \left(1 - y_i(\boldsymbol{w}^\top(\boldsymbol{x}_i - \boldsymbol{u}_i) + b)\right)_+ \,, \tag{2}$$

where $(z)_+ = z$ (0), if $z > 0$ (≤ 0), $\boldsymbol{u}_1, ..., \boldsymbol{u}_\mathsf{n} \in \mathcal{U}$ is a set of bounded perturbations of the training data, and $\mathcal{U}$ is the so-called uncertainty set. This set is defined as:

$$\mathcal{U} \triangleq \{(\boldsymbol{u}_1, \dots, \boldsymbol{u}_\mathsf{n}) | \textstyle\sum_{i=1}^{\mathsf{n}} \|\boldsymbol{u}_i\|^* \leq c\} \,, \tag{3}$$

being $\|\cdot\|^*$ the dual norm of $\|\cdot\|$. Typical examples of uncertainty sets include the ℓ_1 and ℓ_2 balls [22, 17]. The non-robust, regularized optimization problem is formulated as (*cf.* Th. 3 in [22]):

$$\min_{\boldsymbol{w},b} \; c\|\boldsymbol{w}\| + \textstyle\sum_{i=1}^{\mathsf{n}} \left(1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)\right)_+ \,. \tag{4}$$

This means that, if the ℓ_1 norm is chosen as the dual norm characterizing the uncertainty set $\mathcal{U}$, then the optimal regularizer would be ℓ_∞.[2] If the attacker can change only few labels of training samples, label flip attacks can be seen as a sparse ℓ_1 noise affecting the training labels. The optimal countermeasure is therefore to use a ℓ_∞ regularizer to enforce the classifier to give the same importance to all the training samples.

3 Infinity-norm Support Vector Machines

In [8, 15], based on the findings of Xu *et al.* [22] (Sect. 2.3), we have shown that infinity-norm (ℓ_∞) regularization is very effective against *sparse* evasion attacks, *i.e.*, attacks in which the attacker modifies only a small subset of the feature values. The reason is that this regularizer bounds the maximum and the minimum values of the feature weights, *i.e.*, enforcing the SVM to learn more-evenly distributed weights. Under this setting, it is not difficult to see that the attacker is required to manipulate more features to evade detection.

Label-flip attacks can be seen as *sparse* attacks in terms of the influenced training points, since only the labels of few training samples can be manipulated by the attacker. Our idea is thus to exploit ℓ_∞ regularization to enforce more evenly-distributed α weights on the *training data*, similarly to the intuition in [5] to learn more secure SVMs against adversarial label flips. In this work, we obtain this effect by training a (linear) Infinity-norm SVM directly in the kernel space, *i.e.*, using the kernel matrix as the input training data, to learn a discriminant function of the form $g(\boldsymbol{x}) = \sum_{i=1}^{\mathsf{m}} \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) + b$, where $k(\cdot,\cdot)$ is the kernel function, and $\{\boldsymbol{x}_i\}_{i=1}^{\mathsf{m}}$ and $\{\alpha_i\}_{i=1}^{\mathsf{m}}$ are respectively the training samples and their α weights. Under this setting, the α values and the bias b are obtained by solving the following *linear programming* problem:

$$\min_{\boldsymbol{\alpha},b} \qquad \|\boldsymbol{\alpha}\|_\infty + C \textstyle\sum_{i=1}^{\mathsf{m}} (1 - y_i g(\boldsymbol{x}_i))_+ \,. \tag{5}$$

Notably, this approach can be used also with kernels that are not necessarily positive semi-definite (*i.e.*, indefinite kernels).

4 Experimental Analysis

In this section, we first show on a two-dimensional example how the adversarial label-flip attack (ALFA-tilt) affects the decision function of the different SVM classifiers described in the previous sections. Then, to mitigate the fact that the impact of label-flip attacks is strongly data-dependent, as pointed out in [9], we validate our approach on a very large number of real-world datasets, including a case study on PDF malware detection.

[2] The ℓ_1 norm is the dual norm of ℓ_∞, and vice versa.

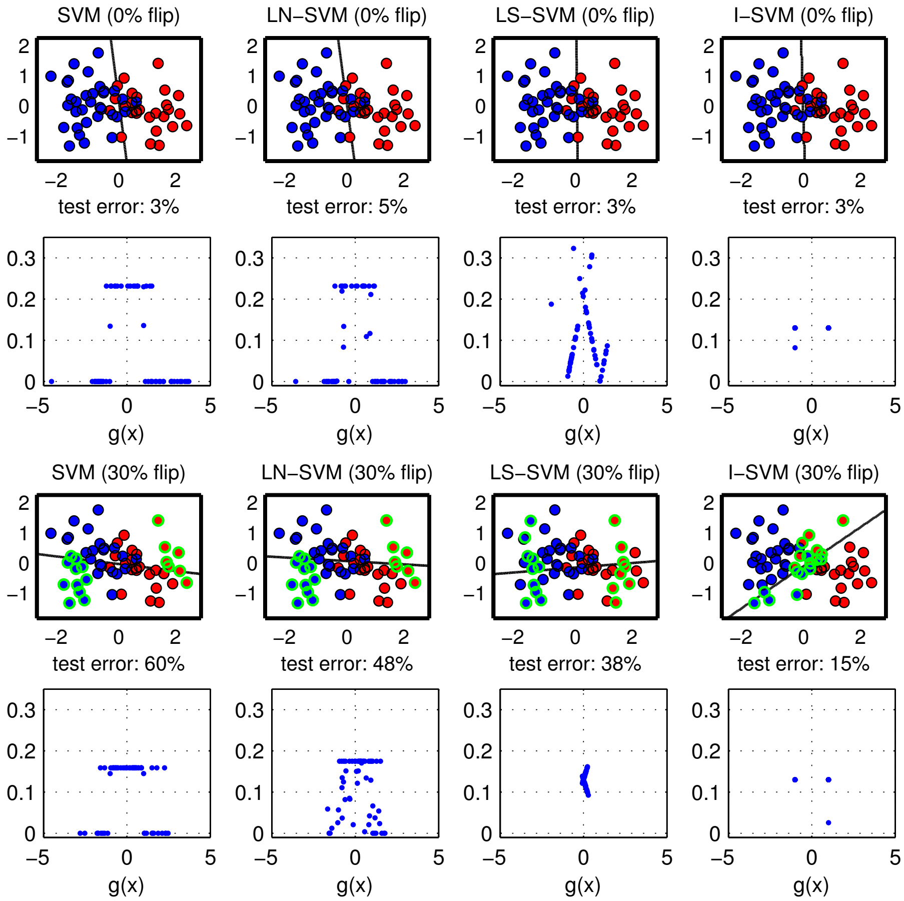

Figure 1: Decision boundaries for the SVM, the LN-Robust SVM (with S=0.1), the LS-SVM, and the Infinity-norm SVM, respectively trained on untainted (first row) and tainted (third row) data. Adversarial label flips are highlight with green circles. For each SVM, we also report the α values assigned to the training samples against the corresponding $g(\boldsymbol{x})$ values.

Two-dimensional Example. We consider here a Gaussian dataset with mean $[y, 0]$ (for class y) and diagonal covariance matrix equal to $\text{diag}([0.5, 0.5])$. We have generated 60 samples for training and 40 for testing, and used the adversarial label-flip attack to flip 18 training labels. We have set $C = 1$ for SVM and LN-SVM, and $C = 0.01$ for LS-SVM and Infinity-Norm SVM. Results are reported in Fig. 1, where one can appreciate how the Infinity-norm SVM retains a higher accuracy under attack, due to the fact that it spreads in a more uniform manner the (absolute) weight values α over the training samples. Note indeed that the decision hyperplane obtained by Infinity-norm SVM under attack, and the corresponding test error, are less affected by the attack.

Real-world data. Here we report the results for 6 datasets downloaded from LibSVM and UCI repositories.[3] Firstly, we have normalized data in $[-1, 1]$ using min-max normalization. Then we have randomly split the data in 5 distinct training and test set pairs, consisting of 60%

[3] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html

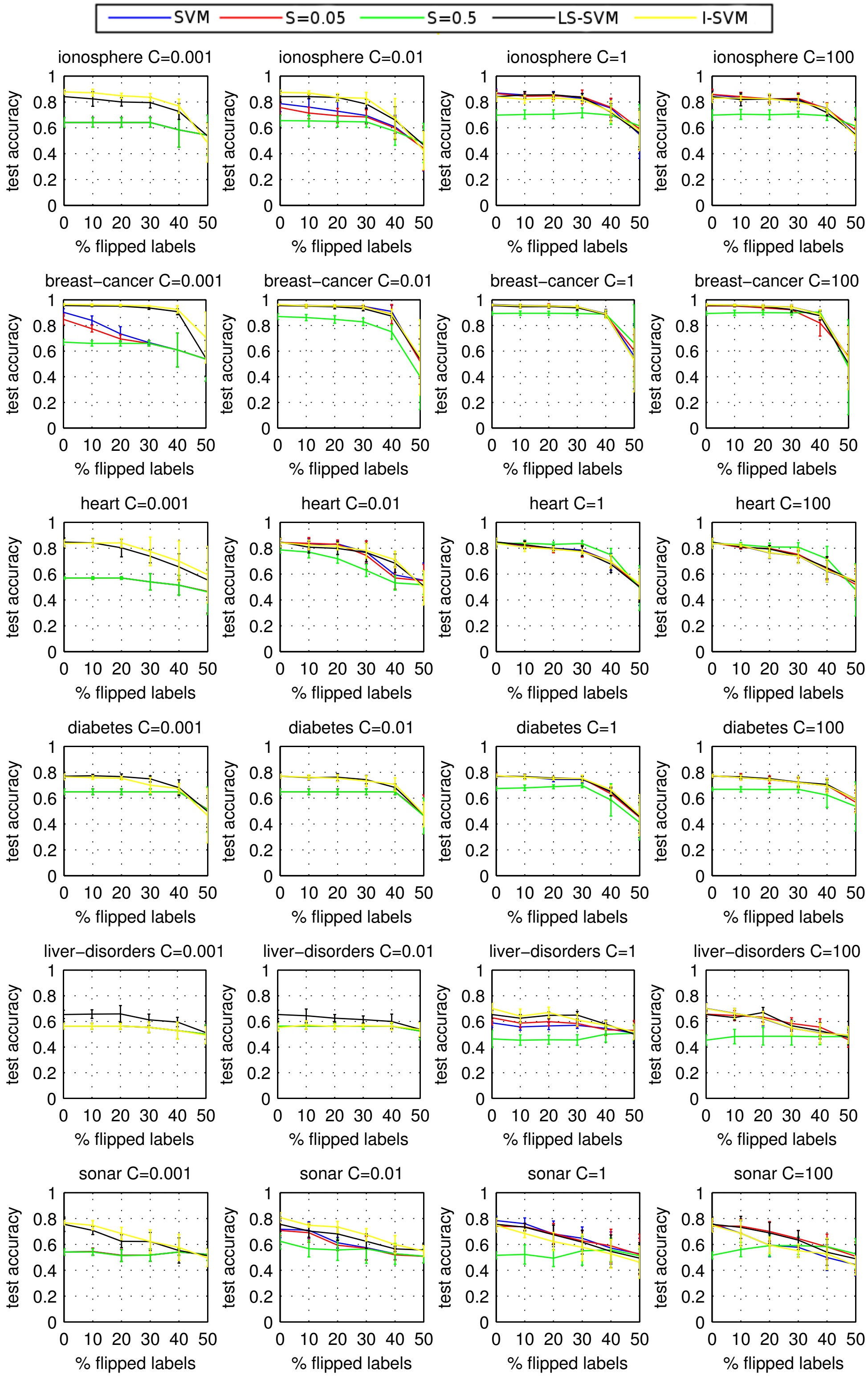

Figure 2: Random label flip attack against SVM, LN-Robust SVM (with S=0.05 e S=0.5), Least-Square SVM, Infinity-norm SVM for different C values on UCI dataset.

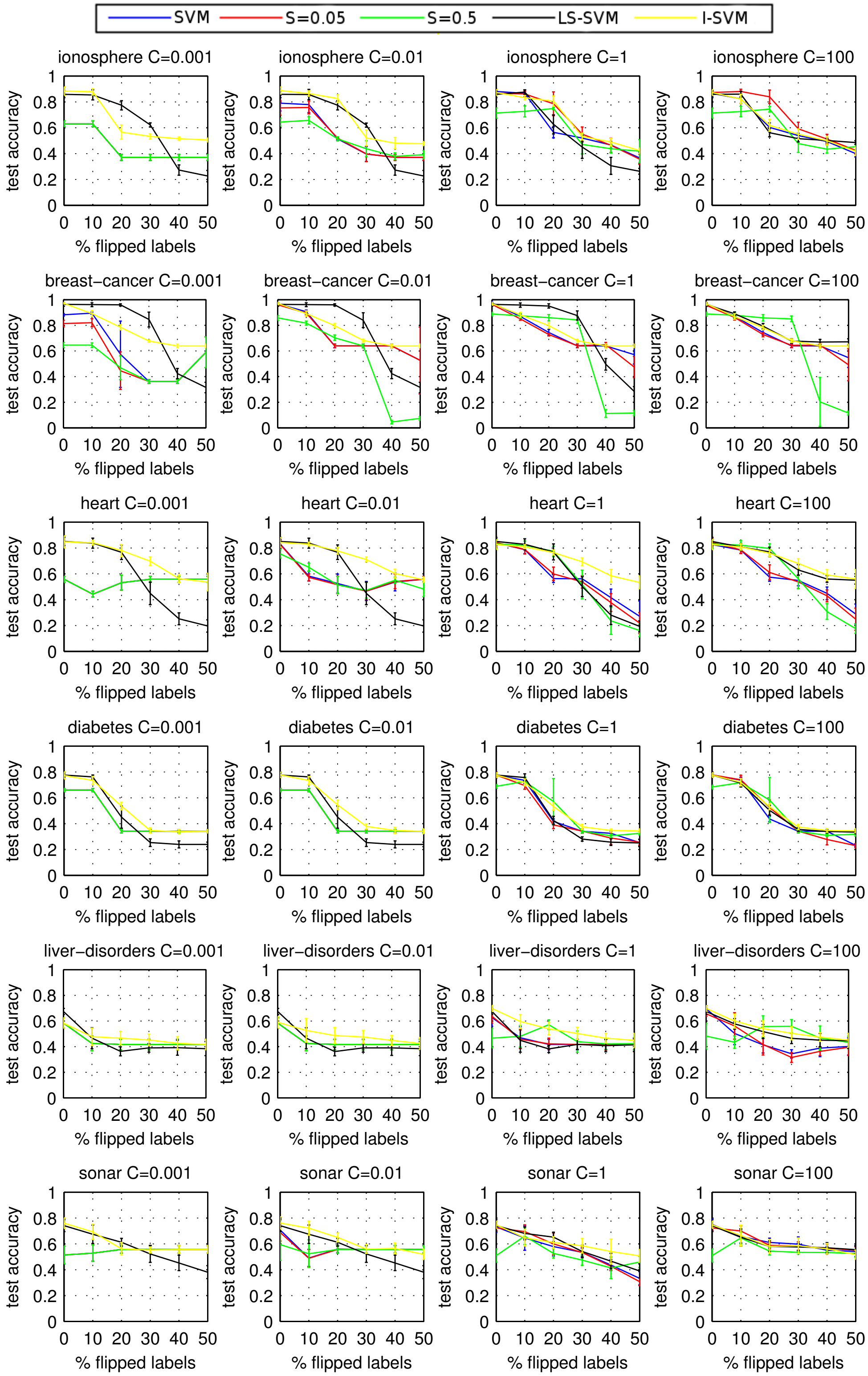

Figure 3: Adversarial label tilt attack against SVM, LN-Robust SVM (with S=0.05 e S=0.5), Least-Square SVM, Infinity-norm SVM for different C values on UCI dataset.

and 40% of the data. The averaged results, for different C values under random and ALFA-tilt attacks are resepctively reported in Fig. 2 and Fig. 3. Notably, the LS-SVM and Infinity-norm SVM attain the best performance for low C values, as the effect of regularization is stronger. These classifiers are nevertheless the most secure under both the random and the ALFA-tilt attack. The performance with the best C for each classifier are dataset-dependent, however Infinity-norm SVM is clearly able to achieve a higher level of security on all the different datasets considered in this evaluation.

PDF Malware Detection. Nowadays PDF is the most used document type due to the fact that presents documents in a independent manner from the operative systems. A PDF document can hosts not only text and images but also JavaScript and Flash scripts. This makes it one of the most exploited vector for convey malware (*i.e.*, malicious software). We have used a dataset called Lux0r [7]. This consists of PDF documents that embeds JavaScript code, collected from different security blogs and antivirus engine. The dataset contains around $12,000$ malicious PDFs and about $5,000$ benign samples. Every PDF is represented by 736 features, each representing the number of occurrences of a specific Javascript function (API call) into the PDF. Each API call corresponds to an action performed by one of the objects that belongs to the PDF. For this experiment, we have used the same normalization and splitting strategy used in our previous experiment on other real-world datasets. The averaged results of this experiment for random and ALFA-tilt attacks are reported in Fig. 4. As for the experiments on the other datasets, we can see that Infinity-norm SVM is able to obtain always good performance and that it has the highest accuracy under the ALFA-tilt attack.

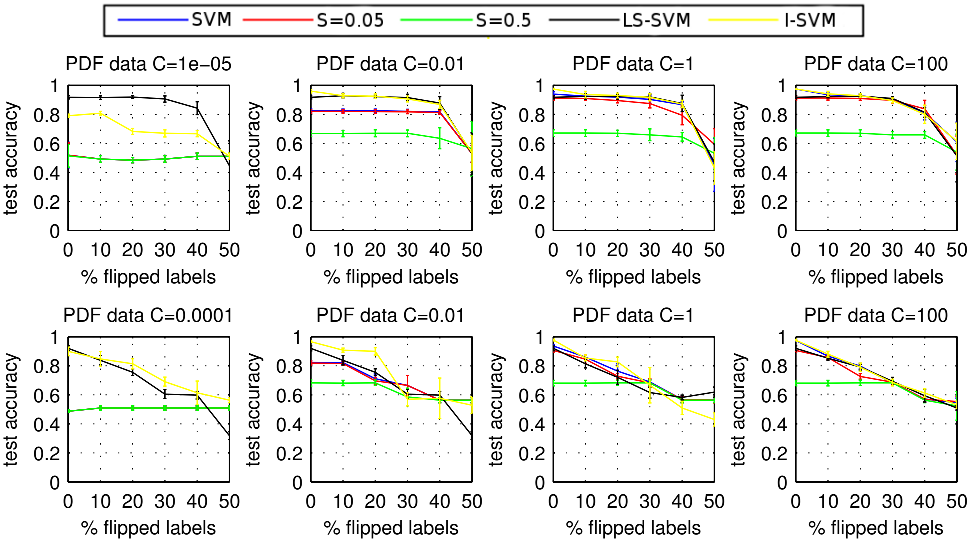

Figure 4: Random (first row) and label tilt attack (second row) against SVM, LN-Robust SVM (with S=0.05 e S=0.5), Least-Square SVM, Infinity-norm SVM for different C values on PDF malware detection dataset.

5 Related Work

Adversarial label flip is a particular case of a more general phenomenon known in the machine-learning literature as *label noise*, as properly explained into a recent survey on this topic by

Frénay *et al.* [9]. As mentioned in Sect. 1, some of the recently-proposed algorithms aim to improve SVM security under random label noise. In [10], Goernitz *et al.* propose a one-class SVM that reduces the influence of outlying data observations during learning. In [14], the authors propose a heuristic approach, named *micro-bagging*, that equalizes the contribution of each training sample, bagging one SVM for each different pair of training samples (each belonging to a different class). Natarjan *et al.* [13] propose a classifier that use a weighted surrogate loss function that represents an upper bound of SVM risk on real data. Their classifier achieves high accuracy also in the presence of a large amount of noise.

Robustness of classifiers against adversarial (worst-case) label flips has been investigated in [12, 6, 5, 20, 21], also proposing some countermeasures to increase classifier security against such attacks.

6 Conclusions and Future Work

Within this work we have investigated the security of different SVMs under adversarial label contamination. We have shown that the sparsity of the SVM α values may be considered a threat for its security in the presence of training data contamination. We have proposed a countermeasure that consists of using an infinity-norm regularizer in kernel space. This proposal is based on more theoretically-sound explanations (in terms of robustness and regularization) than those provided in previous work (mainly based on heuristics and intuition) [5, 21]. We have validated our approach on a large number of real-world datasets, confirming the soundness of the proposed approach. We remark that we have supposed that the attacker has perfect knowledge of the system. Although, in practice, it may be difficult for an attacker to have full knowledge of the targeted system, this is anyway an interesting analysis as it provides an estimate of the maximum performance degradation that the system may incur under attack. Moreover, only relying on security through obscurity (*i.e.*, believing that the attacker is not going to discover some system implementation details) is normally not advocated as a best security practice. Besides considering also limited-knowledge attack scenarios, another interesting future extension of this work may be to investigate the trade-off between robustness to poisoning attacks at training time and evasion attacks at test time, depending on the kind of regularization (and, thus, on the sparsity of the solution). In this respect, it may be interesting to consider novel regularizers that allow one to trade sparsity for classifier security, to tackle computational complexity issues without compromising system security, as also discussed in our recent work for the case of evasion attacks [8].

References

[1] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D. Joseph, and J. D. Tygar. Can machine learning be secure? In *Proc. ACM Symp. Information, Computer and Comm. Sec.*, ASIACCS '06, pages 16–25, New York, NY, USA, 2006. ACM.

[2] B. Biggio, G. Fumera, P. Russu, L. Didaci, and F. Roli. Adversarial biometric recognition : A review on biometric system security from the adversarial machine-learning perspective. *Signal Processing Magazine, IEEE*, 32(5):31–41, Sept 2015.

[3] Battista Biggio, Giorgio Fumera, and Fabio Roli. Pattern recognition systems under attack: Design issues and research challenges. *Int'l J. Patt. Recogn. Artif. Intell.*, 28(7):1460002, 2014.

[4] Battista Biggio, Giorgio Fumera, and Fabio Roli. Security evaluation of pattern classifiers under attack. *IEEE Transactions on Knowledge and Data Engineering*, 26(4):984–996, April 2014.

[5] Battista Biggio, Blaine Nelson, and Pavel Laskov. Support vector machines under adversarial label noise. In *Journal of Machine Learning Research - Proc. 3rd Asian Conf. Machine Learning*, volume 20, pages 97–112, November 2011.

[6] Nader H Bshouty, Nadav Eiron, and Eyal Kushilevitz. Pac learning with nasty noise. *Theoretical Computer Science*, 288(2):255–275, 2002.

[7] Igino Corona, Davide Maiorca, Davide Ariu, and Giorgio Giacinto. Lux0r: Detection of malicious pdf-embedded javascript code through discriminant analysis of API references. In *Proc. 2014 Workshop on Artificial Intelligent and Security Workshop*, AISec '14, pages 47–57, New York, NY, USA, 2014. ACM.

[8] Ambra Demontis, Paolo Russu, Battista Biggio, Giorgio Fumera, and Fabio Roli. On security and sparsity of linear classifiers for adversarial settings. In *Joint IAPR Int'l Workshop on Structural, Syntactic, and Statistical Pattern Recognition*. Springer, Springer, In Press.

[9] B. Frenay and M. Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(5):845–869, 2013.

[10] Nico Görnitz, Anne Porbadnigk, Alexander Binder, Claudia Sannelli, Mikio L. Braun, Klaus-Robert Müller, and Marius Kloft. Learning and evaluation in presence of non-i.i.d. label noise. In *Proc. 17th Int'l Conf. on Artificial Intell. and Statistics, AISTATS, Reykjavik, Iceland, April 22-25*, pages 293–302. JMLR.org, 2014.

[11] L. Huang, A. D. Joseph, B. Nelson, B. Rubinstein, and J. D. Tygar. Adversarial machine learning. In *4th ACM Workshop on Artificial Intelligence and Security (AISec 2011)*, pages 43–57, Chicago, IL, USA, 2011.

[12] Michael Kearns and Ming Li. Learning in the presence of malicious errors. *SIAM J. Comput.*, 22(4):807–837, 1993.

[13] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 1196–1204. Curran Associates, Inc., 2013.

[14] Blaine Nelson, Battista Biggio, and Pavel Laskov. Microbagging estimators: An ensemble approach to distance-weighted classifiers. In *Journal of Machine Learning Research - Proc. 3rd Asian Conf. Machine Learning*, volume 20, pages 63–79, Taoyuan, Taiwan, November 2011.

[15] Paolo Russu, Ambra Demontis, Battista Biggio, Giorgio Fumera, and Fabio Roli. Secure kernel machines against evasion attacks. In *9th ACM Workshop on Artificial Intelligence and Security*. ACM, 2016.

[16] Charles Smutz and Angelos Stavrou. Malicious pdf detection using metadata and structural features. In *Proceedings of the 28th Annual Computer Security Applications Conference*, ACSAC '12, pages 239–248, New York, NY, USA, 2012. ACM.

[17] Suvrit Sra, Sebastian Nowozin, and Stephen J. Wright. *Optimization for Machine Learning*. The MIT Press, 2011.

[18] J.A.K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[19] Gang Wang, Tianyi Wang, Haitao Zheng, and Ben Y. Zhao. Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers. In *23rd USENIX Security Symposium (USENIX Security 14)*, San Diego, CA, 2014. USENIX Association.

[20] Han Xiao, Huang Xiao, and Claudia Eckert. Adversarial label flips attack on support vector machines. In *20th European Conference on Artificial Intelligence*, 2012.

[21] Huang Xiao, Battista Biggio, Blaine Nelson, Han Xiao, Claudia Eckert, and Fabio Roli. Support vector machines under adversarial label contamination. *Neurocomputing, Special Issue on Advances in Learning with Label Noise*, 160(0):53 – 62, 2015.

[22] Huan Xu, Constantine Caramanis, and Shie Mannor. Robustness and regularization of support vector machines. *Journal of Machine Learning Research*, 10:1485–1510, July 2009.