Machine Learning Security Assessment Method Based on Adversary and Attack Methods

Hugo Sebastian Pacheco-Rodríguez[1(✉)], Eleazar Aguirre-Anaya[1(✉)], Ricardo Menchaca-Méndez[2], and Manel Medina-Llinàs[3]

[1] Cybersecurity Laboratory, CIC - Instituto Politécnico Nacional, Av. Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal S/N, Nueva Industrial Vallejo, 07738 México City, Mexico
b190385@sagitario.cic.ipn.mx, eaguirre@cic.ipn.mx

[2] Network and Data Science Laboratory, CIC - Instituto Politécnico Nacional, Av. Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal S/N, Nueva Industrial Vallejo, 07738 México City, Mexico
ric@cic.ipn.mx

[3] Computer Networks and Distributed System (CNDS), Universitat Politècnica de Catalunya, Barcelona, Spain
manel@ac.upc.edu

Abstract. Analytical methods for assessing the security of Machine Learning Systems (MLS) that have been proposed in other researches do not provide compatibility with each other and their taxonomies have become incomplete due to the introduction of new properties of adversarial machine learning. In this sense, we have identified carefully relevant concepts of most prevalent researches about the security assessment of MLS. We propose a novel security assessment method based on the modeling of the adversary and the selection of adversarial attack methods for the generation of adversarial examples related to the also proposed taxonomy. This method provides compatibility with other proposed methods as well as practical guidelines and tools for evaluating machine learning systems. We also introduce the concern for efficient metrics capable of measuring the robustness of MLS to adversarial examples. This research is focused on the empirical evaluation of the security of machine learning systems, rather than on classical performance evaluation.

Keywords: Security · Machine learning · Evaluation

1 Introduction

Research on Adversarial Machine Learning (AML) has grown considerably in recent years and the consequences of unsecured Machine Learning Systems (MLS) have been studied in detail [1–10]. Results of these works are of concern to the scientific community, especially in the field of cybersecurity, because machine learning is being used in different applications to assist in decision making where security is paramount: healthcare, autonomous vehicles, power station operation, military operations, computer security, spam and malware detection, etc.

M. F. Mata-Rivera et al. (Eds.): WITCOM 2020, CCIS 1280, pp. 377–389, 2020.
https://doi.org/10.1007/978-3-030-62554-2_27

Due to the growing concern for the security of machine learning systems, methods have been developed for the evaluation of this type of systems [2, 11, 12]. Each of these methods conceptually defined taxonomies, threat models and attack strategies to assess MLS, including the adversarial properties that were known at the time. Due to the accelerated progress in adversarial machine learning, currently none of them contain a complete taxonomy and threat model that includes the adversarial properties found so far, and therefore do not allow benchmarking between MLS security assessments.

This research complements methods presented in [2, 11, 12] proposing a different organization of the threat model, and introducing the concern for effective metrics capable of measuring the robustness of machine learning systems to adverse examples. It is important to emphasize that security of machine learning is a constant concern, as their security properties have not been completely understood.

Although a defense threat model could be defined [13], this research is limited to the definition of an adversarial threat model.

Section 2 summarizes most relevant researches on adversary threat models in order to design the theoretical adversarial threat model and taxonomy of adversarial attacks. Section 3 provides an overview to perform a security assessment of a machine learning system considering a threat model, the different types of adversarial attack methods, metrics and we also recommend software tools for the generation of adversarial samples.

2 Threat Model and Taxonomy

The adversarial threat model is composed of the goals, capabilities, and knowledge of the adversary, that the MLS to be assessed will face. Conceptually defining the threat model is essential, because it describes the adversary against whom the system intends to defend itself, guiding the evaluation of the machine's learning system.

There are researches [2, 4, 11, 12, 14, 15] where threat models and taxonomies are defined, but often are not compatible between them. In [2, 11, 14] methods are proposed to evaluate MLS, the structure of these methods changes in each one, according to their application. Despite the changes, these investigations share the conceptual definition of the threat model, the taxonomy or the attack strategy. In this research, we propose an organization of the threat model and a general taxonomy for attacks that allows the comparison of MLS security assessments.

We have summarized the predominant concepts in the relevant taxonomies and looked for common features to find a description of each concept compatible with previous work [2, 4, 11, 12, 14, 15]. Concepts presented in Sect. 2.1 are based on taxonomies from the most relevant researches in this research field. The taxonomy for the adversary proposed also defines the organization of the analytical threat model.

2.1 Attack Scenario

Attack scenario must be specified in terms of the conceptual model of the adversary. As well as Biggio et al. [11] model, the following scenario is based on the assumption that, the adversary acts rationally to attain a given goal, according to his/her knowledge of the classifier, and his/her capability of manipulating data.

Adversary Knowledge

The adversary can have different levels of knowledge of the targeted system such as the training data, test data, feature set, learning algorithm, model architecture, model methods or trained parameters/hyperparameters.

Biggio et al. [4] characterized the adversarial knowledge of the targeted system in terms of a space:

$$\Theta = (\mathcal{D}, \mathcal{X}, f, w) \tag{1}$$

Where:

- $\mathcal{D}$: Training data.
- $\mathcal{X}$: Feature set.
- f: Machine learning algorithm, along with the objective function $\mathcal{L}$ minimized during training.
- w: Trained parameters/hyper-parameters.

Depending on the adversary knowledge, one can describe three different type of attacks.

- **White-Box Attacks:** the adversary is assumed to know everything about the targeted system. This setting allows to perform a worst-case evaluation of the security of learning algorithms. It can be characterized as follows:

$$\Theta_{\text{WB}} = (\mathcal{D}, \mathcal{X}, f, w) \tag{2}$$

- **Grey-Box Attacks:** the adversary has partial information about the model. Two main cases are characterized below:

  - Surrogate-Dataset (adversary is assumed to know the feature representation $\mathcal{X}$ and the kind of learning algorithm f):

$$\Theta_{\text{GB-SD}} = \left(\hat{\mathcal{D}}, \mathcal{X}, f, \hat{w}\right) \tag{3}$$

    Where:

    $\hat{\mathcal{D}}$: Surrogate dataset from a similar source.
    $\hat{w}$: Estimated parameters from $\hat{\mathcal{D}}$ (after training a surrogate classifier).

  - Surrogate-Learners (adversary is assumed to know only the feature representation $\mathcal{X}$):

$$\Theta_{\text{GB-SL}} = \left(\hat{\mathcal{D}}, \mathcal{X}, \hat{f}, \hat{w}\right) \tag{4}$$

    Where:

    $\hat{\mathcal{D}}$: Surrogate dataset from a similar source.

$\hat{f}$: Surrogate learning algorithm.
$\hat{w}$: Estimated parameters from $\hat{\mathcal{D}}$ (after training a surrogate classifier).

- **Black-Box Attacks:** the adversary has no knowledge about the model except some components that can be obtained externally. Can be characterized as follows:

$$\Theta_{\mathrm{BB}} = \left(\hat{\mathcal{D}}, \hat{\mathcal{X}}, \hat{f}, \hat{w}\right) \tag{5}$$

Where:

– $\hat{\mathcal{D}}$: Surrogate dataset from a similar source.
– $\hat{\mathcal{X}}$: Surrogate feature set.
– $\hat{f}$: Surrogate learning algorithm.
– $\hat{w}$: Estimated parameters from $\hat{\mathcal{D}}$ (after training a surrogate classifier).

Table 1 shows the three different types of attacks based on the adversary knowledge and their most known components of an MLS respectively.

Table 1. Adversary knowledge

Known component	White-Box	Gray-Box	Black-Box
Training data	X		
Test data	X		
Parameter values	X		
Training method (loss function)	X	X	
Model architecture	X	X	
Feature set	X	X	
Input-output pairings*	X		X
Input-output samples of training data*	X		X

*Input-output samples and pairings are obtained using the targeted machine learning system as an Oracle. The data obtained can be used to train a substitute machine learning model.

Adversary Goals

Adversary Goals are formulated as the optimization of an objective function. Biggio et al. [11] argue that the adversary goal must be defined on the desired security violation, and on the attack specificity. The attack specificity depends on whether an adversary wants to misclassify a targeted or an indiscriminate set of samples. Table 2 summarizes the attack specificity axis.

In [1] Papernot et al. define targeted or indiscriminate attacks depending on whether the adversary aims to cause-specific or generic errors. Because it can cause confusion

Table 2. Attack specificity axis

Attack specificity	Description	Example attack
Targeted [2, 6, 11, 15, 16]	The focus is on a single or small set of target points	• Targeted misclassification • Source-target misclassification
Indiscriminate [2, 6, 11, 15, 16]	Has a flexible goal, that involves a very general class of points, such as "any false negative". Universal adversarial examples are defined here	• Confidence reduction • Misclassification

with the interpretation of targeted and indiscriminate attack specificity Biggio et al. modify their naming convention. The error specificity can thus be: specific or generic. Error Specificity disambiguates the notion of misclassification in multi-class problems. Table 3 summarizes Error Specificity attacks.

Table 3. Error specificity attacks axis

Error specificity attacks	Description	Intends
Specific [4]	The adversary aims to mislead classification but requires the adversarial samples to be misclassified as a specific class	• Maximizes the confidence assigned to the wrong target class, while minimizing the probability of correct classification
Generic [4]	The adversary is interested in misleading classification, regardless of the output class predicted by the classifiers	• Attack will ensure that adversarial sample will no longer classifies correctly as a sample class, but rather misclassified as a sample of the closest candidate class

Desired end security violation (Table 4) relates to the adversary effort to compromise the system. It is important to emphasize that in the case of MLS, integrity is of paramount importance, because attacks on system integrity and availability are closely related in goal and method.

Adversarial Capabilities

It refers to the control that the adversary has on training and testing data. Table 5 summarizes influence axis.

Table 6 summarizes how each author define the threat model in the literature respectively.

Table 4. Security violation adversary axis

Security violation	Description	Attack examples
Integrity [2, 6, 11, 15, 16]	Result in intrusion points being classified as normal (false negatives)	• Confidence reduction • Misclassification • Targeted misclassification • Source-target misclassification
Availability [2, 6, 11, 15, 16]	Cause so many classification errors, both false negatives and false positives, that the system becomes effectively unusable	• Model corruption • Denial of Service
Confidentiality [11, 16]	The adversary obtains information from the machine learning algorithm, compromising the secrecy or privacy of the system users	• Exposure of the model and training data • Membership test • Training data extraction

Table 5. Adversary influence axis

Influence	Description	Attack examples
Causative [2, 6, 11, 15, 16]	Alter the training process through influence over the training data	• Data manipulation • Label manipulation • Input manipulation • Data injection • Logic corruption • Data access
Exploratory [2, 6, 11, 15, 16]	Do not alter the training process but use other techniques, such as test the classifier, to discover information about it or its training data	• Single step (Gradient-based) • Iterative (Gradient-based) • Gradient-free attacks • Extraction • Inversion • Membership inference

As we can see in Table 6, some authors use the terms ‘adversary’ or ‘adversarial’ referring to the ‘attacker’, we will use the term ‘adversary’ and ‘adversarial’ as we consider that it fits better in the context of machine learning security assessment. We also consider the definition of the adversary knowledge involves the definition of the attack surface.

Table 6. Threat model assumptions

Authors	Threat model set assumptions
Barreno et al. [2]	Attacker's goals/incentives Attacker's capabilities
Biggio et al. [11]	Adversarial goals Adversary's knowledge Adversary's capabilities
Carlini and Wagner [14]	Adversary goals Adversary knowledge Adversarial capabilities
Chakraborty and Anirban [15]	Attack surface Adversary capabilities Adversary goals
Papernot et al. [12]	Attack surface Trust model Adversarial capabilities Adversarial goals
Biggio et al. [4]	Attacker's goals Attacker's capabilities

2.2 Attack Strategy

The attack strategy define how the training and test data will be quantitatively modified to optimize the objective function characterizing the adversary goal [11]. Biggio et al. [4] characterized the optimal attack strategy as follows:

$$\mathcal{D}_c^* \in \arg\max_{\mathcal{D}_c' \in \Phi(\mathcal{D}_c)} \mathcal{A}\left(\mathcal{D}_c', \theta\right) \tag{6}$$

Where:

- $\theta \in \Theta$: Adversary knowledge
- $\mathcal{D}_c$: Initial attack samples
- $\Phi\mathcal{D}_c$: Space of possible modifications
- $\mathcal{A}\left(\mathcal{D}_c', \theta\right) \in \mathbb{R}$: Adversary goals objective function
- $\mathcal{D}_c' \in \Phi(\mathcal{D}_c)$: Set of manipulated adversarial examples

3 Security Assessment Method

Most authors proposed security assessments focused on a specific application, classifier, and attack, performing security assessment procedures based on the exploitation of problem knowledge and heuristic techniques. They point to a previously unknown vulnerability or to assess the impact of a known attack on the security of an MLS. Here we

propose an analytical method that complements the existing [4, 11] security assessment methods.

As part of the evaluation model, it is necessary to identify the threat model, in order to illustrate necessary concepts to identify it, the organization of the axes mentioned in Sect. 2 are presented in Sect. 3.1.

Threat model could be interpreted as general guidelines for the security assessment of an MLS. Figure 1 illustrates the assumptions of our proposed threat model. Attack scenario must be defined making assumptions about the adversarial knowledge, adversarial goals and adversarial capability. The definition of the attack strategy is a fundamental part of the model since it attempts to optimize the function that characterizes the adversary goals, we will discuss more about this further.

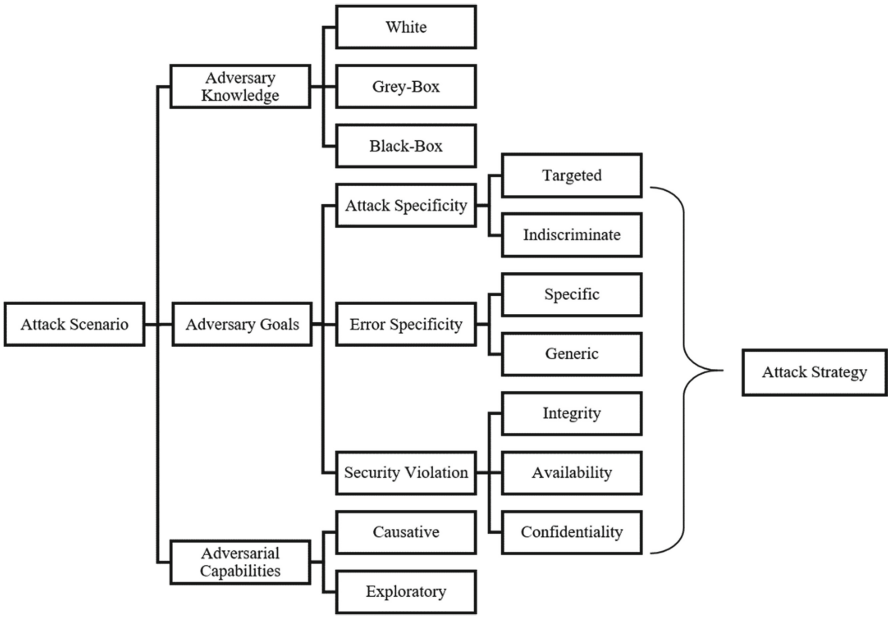

Fig. 1. Threat model for security assessment of machine learning systems.

3.1 Attack Strategy

As we mentioned in Sect. 2.2 the attack strategy must be defined based on the function characterizing the adversary goal. The definition of the attack strategy, the adversary knowledge, and the adversarial capabilities help to define which methods of attack to use. However, it should be mentioned that evaluating MLS with as many methods as possible will provide an even more detailed evaluation.

After defining threat model, attack scenario and attack strategy the adversarial attacks methods must be selected or designed, in Sect. 3.2 we show some state of art methods.

3.2 Adversarial Attacks Methods

Adversarial attack methods should be selected according to the defined threat model to guide the security assessment. Table 7 summarizes the most relevant adversarial attack methods according to our taxonomy proposed in Sect. 2. We consider these attacks as they have shown the best results when vulnerating MLS [15], designer/adversary can select state of the art attacks not mentioned in the table that fits their attack scenario.

Table 7. Most relevant adversarial attack methods for generating adversarial examples

Adversarial attack	Adversarial knowledge	Attack specificity	Attack frequency	Metric
L-BFGS Attack [17]	White-Box	Targeted	Iterative	L_2
Fast Gradient Sign Method (FGSM) [3]	White-Box	Indiscriminate	Single-Step	L_∞
Basic Iterative Method and Least-Likely Class [18]	White-Box	Indiscriminate	Iterative	L_∞
Jacobian-based Saliency Map Attack (JSMA) [1]	White-Box	Targeted	Iterative	L_2
DeepFool [8]	White-Box	Indiscriminate	Iterative	L_1, L_∞
C & W Attack [19]	White-Box	Targeted	Iterative	L_1, L_2, L_∞
Zeroth Order Optimization [20]	Black-Box	Targeted and Indiscriminate	Iterative	L_2
Universal Perturbation [21]	White-Box	Indiscriminate	Iterative	L_1, L_∞
One Pixel Attack [9]	White-Box	Targeted and indiscriminate	Iterative	L_0
Feature Adversary [22]	White-Box	Targeted	Iterative	L_2

We recommend that attack methods be used that fit the assumptions about the adversary knowledge, goals and capabilities, as well as consider the computational cost (attack frequency) and whether the model is gradient-free or not.

In Table 7 we categorized adversarial attack methods according to our taxonomy proposed in Sect. 2, also we introduce under which metric each attack is limited. In Sect. 3.3 we go into detail about this metrics.

3.3 Metrics

Throughout the brief history of adversarial attacks, different metrics have been used to measure the change in the original samples from the adverse samples. Goodfellow and

others used metrics based on L_p norms, however, these types of metrics are not useful for measuring the robustness of an MLS, which is why Weng et al. [23] introduced CLEVER (Cross Lipschitz Extreme Value for nEtwork Robustness), a metric that provides an agnostic measure of attack to evaluate the robustness of any machine learning classifier trained against adversarial examples. In Table 8, we resume metrics used in adversarial settings.

Table 8. Metrics

Metric		Description
Distance metrics	L_0	Measures the number of coordinates i such that $x_i \neq x_i^{'}$. The L_0 distance corresponds to the number of pixels that have been altered in an image
	L_2	Measures the standard Euclidean (root-mean-square) distance between x and x'. Can remain small when there are many small changes to many pixels
	L_∞	Measured the maximum change to any of the coordinates. $\|x - x'\| = \max(\|x_1 - x_1'\|, \ldots, \|x_n - x_n^{'}\|)$
Accuracy		Most publications use accuracy to argue that attacks are effective or in order to evaluate robustness of machine learning

Weng et al. [23] introduce CLEVER an attack agnostic metric to measure lower bound robustness, based on Lipshitz continuity, however, Goodfellow et al. [24] show that CLEVER fails to correctly estimate lower bound robustness, even in theoretical settings. The question of measuring robustness remains open.

We recommend the use of both distance and accuracy metrics, since attacks that remain below the limits of the corresponding L_P norm and obtain high accuracy could be considered effective, and therefore the adversarial robustness of the MLS is considered low.

Derived from the threat model, we can define two types of evaluation methods; one that is directly related to the designer and other to the adversary. Figure 2 briefly illustrates our method for a designer to perform a security assessment, it is important to emphasize that the order cannot be altered.

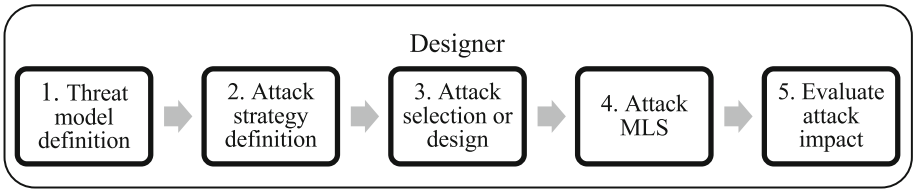

Fig. 2. Designer security assessment method

Figure 3 briefly illustrates our method for an adversary to perform a security assessment, as in designer evaluation method, the order cannot be altered.

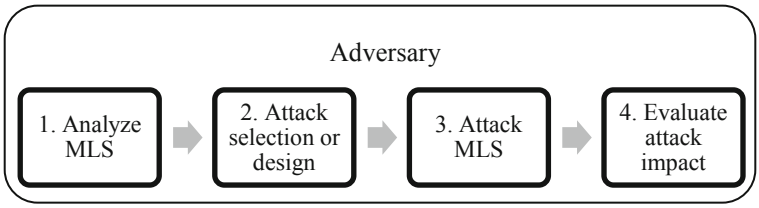

Fig. 3. Adversary security assessment method

4 Discussion

We can observe that the evaluation method proposed is based on modeling the adversary, which allows the designer to anticipate the adversary by identifying threats that the system can face, as well as simulating attacks. The organization of the threat model proposed in Fig. 1 allows us to define the attack scenario and to model the adversary depending on his knowledge, goal, and capability.

On the adversary's side, our threat model will help to analyze the MLS, since he will be able to identify what knowledge, goal and capability he has of the system and then chooses or design an attack method. As a result, we will have a security assessment performed from the adversarial side.

We decided not to include the development of countermeasures as part of the method, as was done in [2, 11] because this research focuses only on the security assessment of MLS. However, we leave open the possibility for the reader to cycle the methods and include the development of countermeasures in order to obtain MLS robust to adversarial attacks.

In Fig. 3.1 we can see in the adversarial goal axis that we include error specificity axis; this is because we find it helpful in evaluating multi-class classifiers. The fact that our method also considers multi-class classifiers makes it a high-level guideline.

5 Conclusions and Future Work

The security assessment method proposed in this paper provides the features necessary to perform security assessments of MLS. Each of the terms used for the conceptual definition of the threat model was compared with its similar, which allowed to choose the organization of the threat model that allows to model the adversary in detail defining assumptions about their goals, knowledge and capabilities. A limitation of the evaluation method for the designer is that it requires a full analysis of the adversary's behavior, which is sometimes difficult and in the case of the evaluation method for the adversary is data-dependent. The unification and update of the previous security assessment methods

as well as the introduction of robustness metrics will allow a more detailed security evaluation of the MLS.

However, there are still open problems, such as analyzing the vulnerabilities of the MLS with respect to adversarial attacks and developing metrics capable of quantifying the robustness of a machine learning system to adversarial examples. These issues will need to be addressed soon to help ensure that the implementation of machine learning systems in adversarial settings is secure.

As future work, we will introduce a defense threat model and defense taxonomy, with the purpose of assessing defense methods for MLS.

Acknowledgements. We would like to thank IPN[1] for allowing us to accomplish this work in the CIC[2]. Pacheco-Rodriguez gratefully acknowledges the scholarship from CONACyT[3] to pursue his master studies.

References

1. Papernot, N., et al.: The Limitations of Deep Learning in Adversarial Settings. Institute of Electrical and Electronics Engineers Inc., November 2015
2. Barreno, M., Nelson, B., Joseph, A.D., Tygar, J.D.: The security of machine learning. Mach. Learn. **81**, 121–148 (2010). https://doi.org/10.1007/s10994-010-5188-5
3. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings (2015)
4. Biggio, B., Roli, F.: Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning (2018)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: Proceedings - IEEE Symposium on Security and Privacy, pp. 39–57 (2017). https://doi.org/10.1109/sp.2017.49
6. Barreno, M., Nelson, B., Sears, R., Joseph, A.D., Tygar, J.D.: Can machine learning be secure? (Invited Talk). Asiaccs **06**, 16–25 (2006). https://doi.org/10.1145/1128817.1128824
7. Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., Hsieh, C.-J.: ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: AISec 2017 - Proceedings of 10th ACM Work. Artificial Intelligence and Security co-located with CCS 2017, pp. 15–26, August 2017. https://doi.org/10.1145/3128572.3140448
8. Moosavi-Dezfooli, S.-M., Fawzi, A., Frossard, P.: DeepFool: a simple and accurate method to fool deep neural networks. In: Proceedings of IEEE Computing Society Conference Computer Vision Pattern Recognition, vol. 2016-December, pp. 2574–2582, November 2015
9. Su, J., Vargas, D.V., Sakurai, K.: One Pixel Attack for Fooling Deep Neural Networks (2017)
10. Athalye, A., Engstrom, L., Ilyas, A., Kevin, K.: Synthesizing robust adversarial examples. In: 35th International Conference on Machine Learning, ICML 2018, vol. 1, pp. 449–468 (2018)
11. Biggio, B., Fumera, G., Roli, F.: Security Evaluation of Pattern Classifiers under Attack (2017)
12. Papernot, N., Mcdaniel, P., Sinha, A., Wellman, M.P.: SoK: Security and Privacy in Machine Learning (2018). https://doi.org/10.1109/eurosp.2018.00035

[1] Instituto Politécnico Nacional (https://www.ipn.mx/) .

[2] Centro de Investigación en Computación (https://www.cic.ipn.mx/).

[3] Consejo Nacional de Ciencia y Tecnología (https://www.conacyt.gob.mx/).

13. Serban, A.C., Visser, J.: Adversarial Examples-A Complete Characterisation of the Phenomenon (2019)
14. Carlini, N., et al.: On Evaluating Adversarial Robustness, February 2019
15. Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., Mukhopadhyay, D.: Adversarial Attacks and Defences: A Survey, September 2018
16. Huang, L., Joseph, A.D., Nelson, B., Rubinstein, B.I.P., Tygar, J.D.: Adversarial Machine Learning (2011)
17. Szegedy, C., et al.: Intriguing properties of neural networks. In: 2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings (2014)
18. Kurakin, A., Goodfellow, I.J., Bengio, S., Brain, G., Openai, I.J.G., Bengio, S.: Adversarial examples in the physical world. In: International Conference on Learning Representations, ICLR (2017)
19. Carlini, N., Wagner, D.: Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods, pp. 3–14, May 2017. https://doi.org/10.1145/3128572.3140444
20. Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., Hsieh, C.-J.: ZOO: Zeroth Order Optimization Based Black-box Attacks to Deep Neural Networks without Training Substitute Models, vol. 17 (2017). https://doi.org/10.1145/3128572.3140448
21. Moosavi-Dezfooli, M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations, vol. 2017-Janua. Institute of Electrical and Electronics Engineers Inc., pp. 86–94 (2017)
22. Sabour, S., Cao, Y., Faghri, F., Fleet, D.J.: Adversarial manipulation of deep representations. In: 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings (2016)
23. Weng, T.W., et al.: Evaluating the robustness of neural networks: an extreme value theory approach. In: 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings (2018)
24. Goodfellow, I.: Gradient Masking Causes CLEVER to Overestimate Adversarial Perturbation Size (2018)