

Adversarial Machine Learning for Cybersecurity and Computer Vision: Current Developments and Challenges

Bowei Xi*

Article Type:

Advanced Review

Abstract

We provide a comprehensive overview of adversarial machine learning focusing on two application domains, i.e., cybersecurity and computer vision. Research in adversarial machine learning addresses a significant threat to the wide application of machine learning techniques – they are vulnerable to carefully crafted attacks from malicious adversaries. For example, deep neural networks fail to correctly classify adversarial images, which are generated by adding imperceptible perturbations to clean images. We first discuss three main categories of attacks against machine learning techniques – poisoning attacks, evasion attacks, and privacy attacks. Then the corresponding defense approaches are introduced along with the weakness and limitations of the existing defense approaches. We notice adversarial samples in cybersecurity and computer vision are fundamentally different. While adversarial samples in cybersecurity often have different properties/distributions compared with training data, adversarial images in computer vision are created with minor input perturbations. This further complicates the development of robust learning techniques, because a robust learning technique must withstand different types of attacks.

*Department of Statistics, Purdue University, Email: xbw@purdue.edu

1 INTRODUCTION

There are two main branches of adversarial machine learning research. One branch actively designs new attacks to defeat existing machine learning algorithms and machine learning based systems. At the same time, the other branch aims to significantly improve the capabilities of machine learning techniques facing attacks.

Attacks against machine learning based systems were observed first in cybersecurity, e.g., (Joseph, Laskov, Roli, Tygar, & Nelson, 2013). One example of adversarial attacks is network intrusion. Attackers actively scan the systems to discover new vulnerabilities in network devices, and to compromise vulnerable hosts/systems. Machine learning techniques were applied to identify the discriminant features of malicious network traffic, e.g., (Sung & Mukkamala, 2003), and classify legitimate network traffic from malicious traffic, e.g., (W. Lee, Stolfo, et al., 1998; Ryan, Lin, & Miikkulainen, 1998; Mukkamala, Janoski, & Sung, 2002; Tsai, Hsu, Lin, & Lin, 2009). As noted early on, e.g., (Householder, Houle, & Dougherty, 2002), attack tools are updated frequently to avoid detection. Hence it is not a trivial task to identify the malicious traffic since the signatures of the malicious traffic

I've been waiting for your reply

Tristan <xxx@verizon.net>

To:xxx@yahoo.com

Feb 6 at 6:41 PM

Hey Alice,

I have been trying to get in touch with you

I am inviting you on board to the system I got going a few weeks ago.

The first participants have seen a reasonable amount of success.

Your invite is attached, the password is 815452

! Attachments cannot be downloaded.

[Learn More](#)

Figure 1: A spam email sent on Feb. 06 2019, and received in Yahoo Spam folder

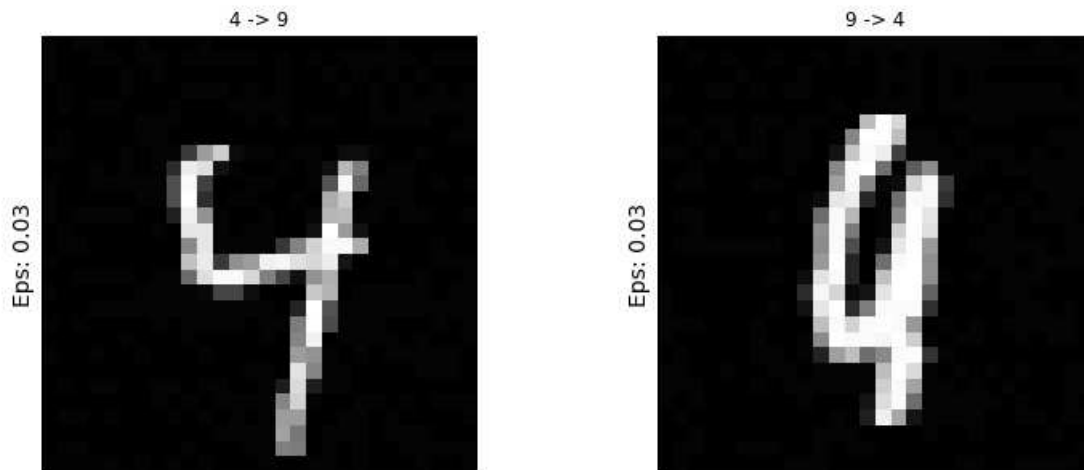


Figure 2: Two adversarial samples. Left: A handwritten 4 is misclassified as 9; Right: A handwritten 9 is misclassified as 4.

change suddenly. Another example is spam. Spam filters on email servers serve as classifiers that identify spam emails. Spam filters either directly block the spam emails or place them in a separate spam folder. Although people have been combating spam emails for over 20 years, e.g., (Cranor & LaMacchia, 1998), spam emails are still a nuisance today. Spammers frequently change how spam emails are composed to avoid detection. Spammers in early 2000s mis-spelled the words that could easily be identified, such as writing “Pharmaceutical” as “Phaxrrmaceutical”, and inserted good words, i.e., words often observed in legitimate emails, into spam emails (Lowd & Meek, 2005). Figure 1 shows a spam email received in Yahoo Spam folder in Feb, 2019. The entire content of this recent spam email is similar to legitimate emails.

Recently adversarial samples are designed in computer vision to break deep neural networks (DNN). Although DNNs are capable of performing complex tasks, they are shown to be vulnerable to attacks. Adversarial images created with minor perturbations are misclassified by DNN, as illustrated in Figure 2.

Adversarial attacks against DNN draw significant attention due to the wide use of DNN in critical tasks, such as autonomous driving systems and partly automated vehicles. For example, Tesla has a on-board computer that “runs the new Tesla-developed neural net

for vision, sonar and radar processing software Built on a deep neural network, Tesla Vision deconstructs the car’s environment at greater levels of reliability than those achievable with classical vision processing techniques.”¹ On the other hand, in one Tesla fatal crash, “The car veered off the road due to limitations of the Tesla Autopilot vision systems processing software to accurately maintain the appropriate lane of travel.”² Such high profile failure of the hugely popular deep learning technique shows the importance of adversarial machine learning area. Researchers must fully understand the vulnerabilities of the learning techniques and consequently robustify the existing learning techniques for them to be used reliably in real life tasks.

There are different types of adversarial samples. Adversarial images are samples with minor perturbations added to the clean images, whereas adversarial samples in cybersecurity have very different properties/distributions compared with the training samples. Furthermore, adversaries seek to learn information about the training data points with sensitive information through a learning model, causing a privacy leakage.

Hence machine learning techniques need to significantly improve their generalization capability to properly handle the adversarial samples created with only minor perturbations, as observed in computer vision. Learning techniques also need new capabilities to identify adversarial samples that change quickly and have very different properties, as observed in cybersecurity. Meanwhile, learning models with complex structure, such as DNNs, may still suffer privacy leakage. It is an urgent and important task to develop secure and robust machine learning techniques that can withstand different types of malicious attacks. Currently there are different approaches. We observe ideas are converging from different communities, such as cybersecurity, computer vision, theoretical machine learning, and artificial intelligence, to achieve this goal.

In this review article, we introduce different types of malicious attacks, the existing strategies to robustify and secure machine learning techniques, and the challenges faced in adversarial machine learning area. The rest of the paper is organized as follows. Section 2

¹<https://www.tesla.com/autopilot> (last verified on 03/01/2020)

²https://www.washingtonpost.com/local/trafficandcommuting/deadly-tesla-crash-tied-to-technology-and-human-failures-ntsb-says/2020/02/25/86b710bc-574d-11ea-9b35-def5a027d470_story.html (last verified on 03/01/2020)

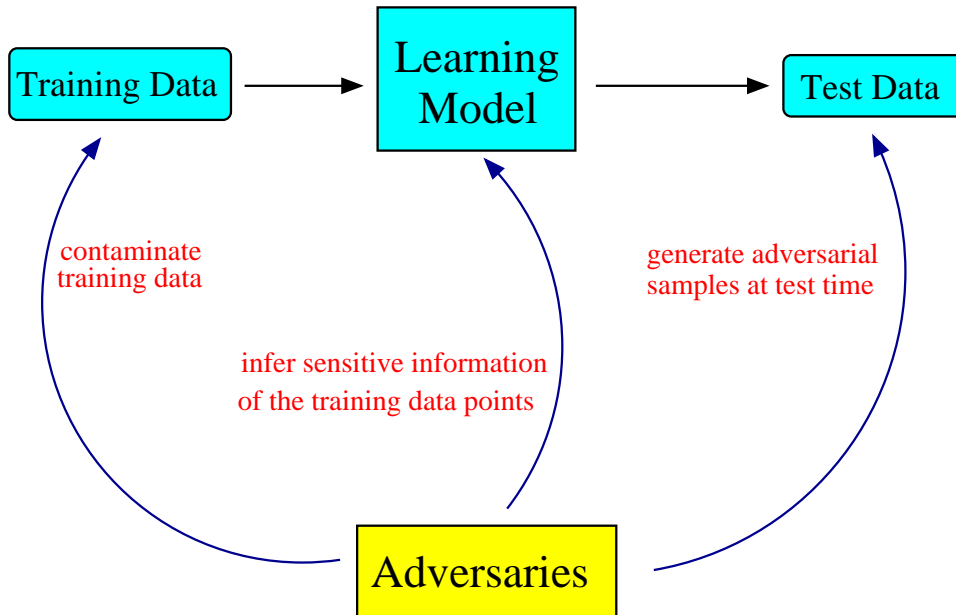


Figure 3: Illustration of poisoning attack, evasion attack, and privacy attack

discusses the adversarial attacks; Section 3 introduces the current defense approaches; Section 4 discusses the challenges that remain to be addressed and concludes this article.

2 Adversarial Attacks

Machine learning techniques are widely used in different applications. We observe many exciting developments of new techniques, such as the success of DNN in computer vision tasks. Unfortunately there are a growing number of vicious attacks against learning models too. In this section we introduce three common forms of attacks. As illustrated in Figure 3, adversaries can contaminate the training data to make a classifier ineffective (poisoning attack), create adversarial samples at test time to evade detection (evasion attack), and infer sensitive information about the training data through a learning model (privacy attack).

2.1 Poisoning Attack

Poisoning attacks aim to contaminate the training dataset and cause a learning model to make costly mistakes at the test time. Poisoning attacks, as well as evasion attacks, can

be either targeted or non-targeted. A targeted attack causes a learning model to make a specific mistake, e.g., misclassifying a certain type of legitimate emails. A non-targeted attack reduces the overall accuracy of a learning model. Earliest examples of poisoning attacks are those against a spam filter (Nelson et al., 2008; Barreno, Nelson, Joseph, & Tygar, 2010; Huang, Joseph, Nelson, Rubinstein, & Tygar, 2011), against an anomaly detection algorithm (Laskov & Kloft, 2009; Kloft & Laskov, 2010), and against intrusion detection systems (Corona, Giacinto, & Roli, 2013).

- (Nelson et al., 2008; Barreno et al., 2010; Huang et al., 2011) considered a scenario where a spam filter were periodically retrained, and spammers sent emails containing words frequently observed in legitimate emails. Although the emails from spammers were labeled correctly as spam, by including them in the retraining process, they caused a spam filter to misclassify legitimate emails.
- (Laskov & Kloft, 2009; Kloft & Laskov, 2010) provided a quantitative strategy to design and analyze the effectiveness of poisoning attacks against centroid based anomaly detection algorithm, which used the distance from the mean of the training data to define anomalies. The effectiveness of a poisoning attack was measured by the relative distance of how much the center was moved by injecting new attack data points into the training dataset. Upper bounds were established under partial control and full control of the training data.
- Adversarial attacks in cybersecurity are generated with a good understanding of the computer systems and their security policies. Poisoning attack is one of the six attack categories against intrusion detection systems, as discussed in (Corona et al., 2013).

Meanwhile we also witness poisoning attacks against specific learning models, including support vector machine (SVM), LASSO and DNN.

- (Biggio, Nelson, & Laskov, 2012) introduced an algorithm to generate a poisoning attack sample x^* against SVM in a two class scenario. Starting from a randomly selected sample with flipped label, the algorithm computed the gradient of the hinge

loss function $\frac{\partial L(x^*)}{\partial w}$ and iteratively searched for a poisoning attack sample to maximize SVM’s validation error.

- (Xiao et al., 2015) introduced poisoning attack against penalized regression models, including LASSO and elastic net. The algorithm generated a poisoning attack sample x^* through maximizing the objective function, $\max_{x^*} \left(\frac{1}{n} \|\hat{Y} - \hat{W}'X - \hat{b}\|_2^2 + \lambda \times p(W) \right)$, where $p(W)$ is the corresponding penalty term for LASSO or elastic net. The generated attack samples rendered the associated feature selection process unstable. (Xiao et al., 2015) showed LASSO selected nearly random variables when less than 5% of generated poisoning attack samples were added to the training data.
- (Gu, Dolan-Gavitt, & Garg, 2017) discussed the danger of having a “backdoor” in a DNN, a.k.a. Trojan attack. A DNN trained by a malicious adversary behaved normally for most of the samples, and produced an error only when a backdoor trigger was present. For example, a sticker posted on a stop sign caused it to be misclassified.

2.2 Evasion Attack

Evasion attacks are the most popular form of attacks. Often malicious adversaries do not have access to the training process. They generate adversarial samples to cause misclassification by a classifier or to evade detection by a learning algorithm at the test time. Spam emails, malwares, and network intrusion traffic observed at the test time, e.g., (Šrndić & Laskov, 2013; Grosse, Papernot, Manoharan, Backes, & McDaniel, 2017), are examples of evasion attacks in cybersecurity.

More recently, in computer vision, many evasion attack algorithms are developed to generate adversarial samples that lead to misclassification by DNNs, starting with the first paper published in 2014 (Szegedy et al., 2014). Attacks against DNN can be either digital or physical. A digital attack directly add minor perturbation to the input to a DNN, i.e., an image, whereas a physical attack creates minor perturbation on a physical object which leads to DNN’s failure to properly recognize the physical object. There are few physical attacks compared with a large number of digital attacks. We focus on digital attacks in this section and discuss physical attacks in Section ??.

Several survey articles are published providing the detailed timeline of adversarial attacks against DNNs, e.g., (Yuan, He, Zhu, & Li, 2019; Akhtar & Mian, 2018; Biggio & Roli, 2018). In 2017, a NIPS competition organized by Google Brain attracted many researchers to develop new adversarial attack algorithms and effective defenses against attacks. The winning teams’ results are documented in (Kurakin et al., 2018). Due to the limited space, here we provide a summary list of digital evasion attacks against DNNs ³.

Attacks can be categorized based on attacker’s knowledge of the learning model. White box attack refers to attacker having a complete knowledge about the DNN under attack, including the model structure and all the parameter values. Black box attack means attacker does not have access to a DNN’s internal structure. Often in black box attack, an attacker sends queries to probe the target DNN. The attacker then uses the queries labeled by the target DNN to train a substitute model and generates adversarial samples against the substitute model. Majority of the digital attacks against DNN are white box attacks. Below we introduce the most notable white box attacks and black box attacks.

- First we introduce the white box attacks.
 - **L-BFGS**: Vulnerability of DNN was first reported in (Szegedy et al., 2014): Adding certain non-random imperceptible perturbation δ to an image x can cause a DNN C to misclassify the adversarial image $x^a = x + \delta$. The paper also observed the transferability of adversarial images – the specific perturbation δ can cause a different DNN C^* to misclassify the same adversarial image x^a . For a given image x and a target class label t , δ was obtained by solving a box constraint optimization problem as follows.

$$\min \|\delta\|_2, \quad s.t. \quad C(x + \delta) = t, \quad x + \delta \in [0, 1]^m$$

The L-BFGS method was used to find an approximate solution.

- **Fast Gradient Sign Method (FGSM)** and its variations: Let $J(x, y)$ be the cross-entropy cost function and y be the true label of a clean image x . Using the

³Code for many attack algorithms and some defenses can be found on GitHub, e.g., (*IBM Adversarial Robustness Toolbox (ART v0.8.0)*, n.d.; *Cleverhans adversarial examples library*, n.d.).

sign of the gradient of the cost function, (I. J. Goodfellow, Shlens, & Szegedy, 2015) generated adversarial image x^a as

$$x^a = x + \epsilon \operatorname{sign}(\nabla_x J(x, y)).$$

Compared with L-BFGS, FGSM lowered computation cost to generate an adversarial image. FGSM is a one-step attack. There are variations of FGSM, including improved one-step attacks or iterative attacks. For example, (Rozsa, Rudd, & Boulton, 2016) produced more diverse adversarial images by using the scaled gradient instead of the sign of the gradient. (Tramèr et al., 2018) proposed a small random perturbation in the one step attack. (Kurakin, Goodfellow, & Bengio, 2017b) suggested adversarial images can be generated using the predicted labels in FGSM instead of the true labels. Instead of a simpler one step attack, adversarial samples can be generated by iteratively following the direction of the gradient while clipping the generated image to stay inside the ϵ ball of the clean image x , i.e., the Basic Iterative Method (BIM) in (Kurakin, Goodfellow, & Bengio, 2017a). Projected Gradient Descent (PGD) attack (Madry, Makelov, Schmidt, Tsipras, & Vladu, 2018) showed that BIM with random starting points in the ϵ ball yielded stronger adversarial samples. (Dong et al., 2018) further used momentum in iterative FGSM to increase attack strength and maintain transferability of adversarial images. The update with momentum is the following.

$$h_{t+1} = \alpha h_t + \frac{\nabla_x J(x_t, y)}{\|\nabla_x J(x_t, y)\|_1}, \quad x_{t+1} = x_t + \epsilon \operatorname{sign}(h_{t+1}).$$

To attack an ensemble of DNNs, (Dong et al., 2018) computed the cross-entropy $J(x, y)$ by averaging the logits of DNNs (i.e., output from the layer before softmax) in the ensemble, then followed the direction of the gradient with a momentum to attack the ensemble.

- **Carlini and Wagner (C&W)**: C&W attack (Carlini & Wagner, 2017b) modified the objective function and used a different optimizer compared with L-BFGS attack. Their L_2 attack is sufficiently strong to bypass a number of detection and defense methods. They solved the following box constraint optimization problem

to find an adversarial perturbation δ .

$$\min_{\delta} (\|\delta\|_2 + c \max(\max_{l \neq o} (Z(x + \delta)_l) - Z(x + \delta)_o, -k)), \text{ s.t. } x + \delta \in [0, 1]^m,$$

where $Z(x)_o$ is the output of the softmax for class o . After a change of variable, $\delta = \frac{1}{2}(\tanh(w) + 1) - x$, Adam optimizer was used to search for w . Following the line of C&W attack, Elastic-net attacks to DNNs (EAD) in (P.-Y. Chen, Sharma, Zhang, Yi, & Hsieh, 2018) added an elastic net type of penalty term in the optimization problem to search for adversarial perturbations. EAD’s L_1 attack achieved comparable performance as the C&W L_2 attack.

- **Jacobian-based Saliency Map Attack (JSMA)**: JSMA computed the Jacobian of either the logits as in (Papernot, McDaniel, Jha, et al., 2016) or the outputs of the softmax as in (Papernot, McDaniel, Wu, Jha, & Swami, 2016) for a clean image x , and built a saliency map. Based on the saliency map, JSMA iteratively chose a pixel to change in each step to increase the likelihood of the target class for an adversarial image.
 - **DeepFool**: (Moosavi-Dezfooli, Fawzi, & Frossard, 2016) introduced an iterative attack algorithm to efficiently search for adversarial samples. At each iteration, a classifier $C(x_t)$ was linearized at the current point x_t and an update was computed as the scaled gradient with respect to the linearized classifier. (Moosavi-Dezfooli, Fawzi, Fawzi, & Frossard, 2017) leveraged the DeepFool attack to find universal adversarial perturbations for almost all the clean images to fool a classifier.
 - **Adversarial Transformation Networks (ATN)**: (Baluja & Fischer, 2018) trained a neural network that modified a clean image into an adversarial sample to fool a target network or an ensemble of networks. ATN was trained by minimizing a loss function which balanced the loss on the input image x and the output of the target network. ATN can be used for both white box and black box attack.
- Next we introduce several black box attacks.
 - **Zeroth Order Optimization (ZOO)**: ZOO (P.-Y. Chen, Zhang, Sharma, Yi,

& Hsieh, 2017) changed the loss function in C&W attack to

$$\max(\max_{l \neq o}(\log F(x)_l - \log F(x)_o), -k),$$

where $F(x)$ is the output of a DNN. ZOO avoided directly computing the gradient. Instead ZOO approximated the gradient using the symmetric difference quotient method, with an increased computation cost. The knowledge of a DNN's network structure was not required to approximate the gradient.

- **OnePixel:** OnePixel attack (Su, Vargas, & Sakurai, 2019) changed the value of only one pixel of a clean image to fool a DNN with differential evolution using only the predicted outputs from a DNN without knowing its network structure. OnePixel showed DNN is even vulnerable to very low dimension attack with limited information.
- (Papernot et al., 2017) used the probing approach to train and generate adversarial images against a local substitute model. With 800 queries sent to Amazon Web Services and Google Cloud Prediction, most of the adversarial samples were misclassified by the target models hosted by Amazon and Google. (Y. Liu, Chen, Liu, & Song, 2017) generated a targeted attack against k networks in the white box fashion, and showed the adversarial images can transfer to an additional black box network. (Ilyas, Engstrom, Athalye, & Lin, 2018) further developed a black box attack under limited queries and partial output knowing only the top k class probabilities. Unlike previous attacks against image level classifier, (X. Liu et al., 2019) developed a black box attack against state-of-the-art object detectors.

2.3 Privacy Attack

Protecting data confidentiality has long been an important research area. The research on privacy leakage and privacy preserving techniques preceded adversarial machine learning. Mostly it remains a separate research area. This review article includes privacy attack as a third attack because the very recent privacy attacks targeting DNNs, which points to a different type of DNN vulnerability. Recently there are two notable privacy attacks – model

inversion attacks and membership inference attacks – designed to obtain sensitive training data information based on the outputs from complex learning models such as DNNs. Similar to evasion attacks, privacy attacks can be either white box or black box, i.e., with or without access to the internal structure and the parameters of the target learning model. A brief review of different privacy preserving approaches as well as the recent attempts to defend DNNs from privacy attacks are in Section 3.2.

- **Model Inversion Attacks:** (Fredrikson, Jha, & Ristenpart, 2015) designed two model inversion attacks against face recognition systems. The first attack can reconstruct a face image based on the person’s unique label produced by a face recognition system. The second attack can obtain a clean image from a blurred image through attacking a system, thus obtaining the identity of the victim. (X. Wu, Fredrikson, Jha, & Naughton, 2016) provided a formal description of model inversion attacks in black box scenario and white box scenario. It also suggested model invertibility was related to the influence/stable influence of Boolean functions.
- **Membership Inference Attacks:** A basic membership inference attack discovers whether a data point belongs to a learning model’s training data, given only black box access to the model (Shokri, Stronati, Song, & Shmatikov, 2017). They used several shadow models trained on synthetic or noisy data to determine whether a data point belonged to the training data or not. They successfully designed membership inference attacks against Google Prediction API and Amazon ML. (Melis, Song, De Cristofaro, & Shmatikov, 2019) showed that when several trained models periodically shared updates, the process can be exploited to infer whether a data point was used in training. They also suggested DNN’s complex structure made it vulnerable to membership inference attacks under collaborative learning setting.

3 Robust and Secure Learning Strategies

The concept of malicious noise and robustness of learning algorithms including neural networks were studied as early as 1985. (Valiant, 1985) introduced a distribution free model, i.e.,

the probably approximately correct (PAC) model, which can tolerate a low rate of malicious noise. (Kearns & Li, 1993) established that the upper bound of malicious error rate which can be tolerated by a learning algorithm A with accuracy $1 - \epsilon$ is $\frac{\epsilon}{1+\epsilon}$. (Kearns & Li, 1993) further constructed perceptron based PAC algorithms with malicious noise tolerance. Then (Servedio, 2001) built perceptron based PAC algorithms with much higher level of malicious noise tolerance. (Teo, Globerson, Roweis, & Smola, 2008) developed learning algorithms that incorporated invariant transformations as a form of robust learning procedure.

Meanwhile researchers hope to understand the baffling phenomenon of adversarial perturbations that cause DNN to make mistakes. (Szegedy et al., 2014) suggested the highly non-linear nature of DNN made it vulnerable, whereas (I. J. Goodfellow et al., 2015) mentioned it was the locally linear structure which made DNN vulnerable. It is generally agreed that DNNs with larger capacity in terms of number of parameters are more robust, e.g., (Kurakin et al., 2017b; Madry et al., 2018).

In addition, recently researchers proposed various metrics to evaluate DNN’s robustness and predictive uncertainty. (Biggio, Fumera, & Roli, 2014) developed an algorithm to empirically evaluate a classifier’s performance under simulated attacks. (Weng et al., 2018) proposed a robustness metric for DNN based on extreme value theory, i.e., Cross Lipschitz Extreme Value for nEtnetwork Robustness (CLEVER). (Moosavi-Dezfooli et al., 2016) used the estimated minimum amount of perturbations needed for an attack to fool a DNN as a measure of DNN’s robustness. (Shafahi, Huang, Studer, Feizi, & Goldstein, 2019) investigated the relationship between dimensionality and the robustness of a classifier.

(Lakshminarayanan, Pritzel, & Blundell, 2017) used deep ensembles based on random initializations and adversarial training to obtain predictive uncertainty estimates. (Gal & Ghahramani, 2016) used Monte Carlo dropout on test samples to estimate predictive uncertainty. (Malinin & Gales, 2018) used a Bayesian approach to evaluate DNN uncertainty caused by distribution shift between training and test samples.

It is an ongoing effort to secure machine learning models. Here we introduce several approaches to improve the performance of learning techniques in adversarial environment.

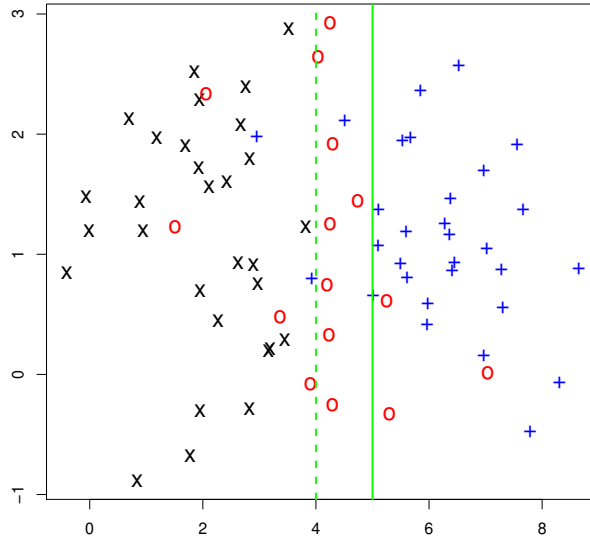


Figure 4: A standard classification boundary (dashed line) vs. a conservative boundary (solid line)

3.1 Game Theoretic Approaches

Game theory provides a useful tool to model and understand the interaction between the defender and the adversaries. Facing adversarial attacks, one strategy is to explore the robustness and accuracy trade-off when building learning algorithms. An equilibrium solution can be utilized to construct robust learning algorithms, which make conservative decisions for the current data but have persistent good performance against potential future attacks. The learning algorithm's decision boundary depends on the game that is used to model the interaction between defense and attacks. The minimax solution in a zero-sum game leads to a conservative strategy when the adversarial samples' property changes abruptly, whereas a mixed equilibrium strategy suggests a randomized defense against minor adversarial perturbations. Figure 4 is a conceptual plot illustrating a conservative decision boundary. Sometimes a minimax solution can be too conservative. Instead, a Stackelberg game offers a less conservative equilibrium solution. To search for an equilibrium is often computationally expensive. However, game theory offers important insights towards resilient algorithms against malicious adversaries. Next we discuss zero-sum game and Stackelberg game based

robust learning solutions.

- A two player zero-sum game where players take simultaneous actions is formulated as follows. Let D be the defender and A be the adversary. Let U_D and U_A be their corresponding utilities. Let (d, a) be a pair of strategies, for the players D and A respectively. In a zero-sum game, $U_A(d, a) = -U_D(d, a) = U(d, a)$. For example, d can be a classifier built by defender D and a can be a transformation of malicious objects by the adversary A . The Nash equilibrium is the optimal solution in a game where no player has the incentive to unilaterally change its strategy. The minimax solution (d_e, a_e) is a Nash equilibrium in a two player non-cooperative game, defined as follows.

$$U(d_e, a_e) = \min_d \max_a U(d, a).$$

Hence at the equilibrium the defender D wants to minimize its worst case loss.

- (Dalvi, Domingos, Mausam, Sanghai, & Verma, 2004) applied a game theoretic approach to spam detection with Naive Bayes as the underlying classifier. The framework allowed a classifier to automatically adjust its classification rule facing evolving manipulations from an adversary.
 - (Lanckriet, Ghaoui, Bhattacharyya, & I., 2002) studied the binary classification problem. For a class, an ellipsoid shaped region was defined using the mean and the variance-covariance matrix. They assumed arbitrary sets of data points can be drawn from their respective ellipsoids. A robust linear classifier was proposed based on a minimax solution given this bounded data uncertainty.
 - (Globerson & Roweis, 2006) studied a two player game where the adversary deleted features that led to the maximum drop of a classifier accuracy and the defender built a robust classifier against malicious feature removal. The minimax solution was proposed with SVM as the underlying classifier. In (Dekel, Shamir, & Xiao, 2010), missing or corrupted features caused by an adversary can vary from one instance to another.
- A Stackelberg game where two players S_L and S_F take sequential actions is formulated as follows. Let S_L be the leader with utility U_L . Let S_F be the follower with utility

U_F . The leader S_L chooses a strategy l first. After observing the leader’s action, the follower S_F chooses its strategy f . The game’s subgame perfect equilibrium (l^e, f^e) can be written as follows.

$$f_l = \operatorname{argmax}_f S_F(l, f), \quad (l^e, f^e) = \operatorname{argmax}_l S_L(l, f_l).$$

Compared with zero-sum game and minimax solution, a Stackelberg game allows each player to maximize its own utility and offers a less conservative solution.

- (Kantarcioğlu, Xi, & Clifton, 2011) modeled adversarial classification as a Stackelberg game where the defender was the follower in the game. The adversary first transformed the objects under its control knowing the action reduced the utilities of the malicious objects. Then the defender built a classifier minimizing the overall misclassification cost. For continuous variables, an approximate equilibrium solution was found by discretizing the variables and using a linear programming approach. A classifier’s equilibrium performance was examined which indicated its long term success or failure facing a malicious adversary.
- (W. Liu & Chawla, 2010) modeled the interaction between the defender and the adversary as a constant-sum Stackelberg game with convex loss. The equilibrium was the solution of a maxmin problem, solved via trust region methods. The game was repeated with the adversary using different data transformations and the defender adjusting the classification rule.
- (Brückner & Scheffer, 2011) let the defender be the leader in a Stackelberg prediction game. The equilibrium was the solution of a bilevel optimization problem, which can be solved by sequential quadratic programming. (Brückner & Scheffer, 2011) showed that the Stackelberg game with the worst case loss given the defender chose a hinge loss led to a SVM for invariances, and a Stackelberg game with linear loss led to a regular SVM. On the other hand, a Stackelberg game with logistic loss was the most robust against attacks.
- (Zhou & Kantarcioğlu, 2016) studied a single-leader-multiple-follower nested Stackelberg game with one defender and different types of adversaries. The game was

formulated as multiple lower level Stackelberg games and one upper level Bayesian Stackelberg game. The defender played a mixed equilibrium strategy, which can be found by solving multiple single-leader-single-follower games with probabilities determined by the Bayesian Stackelberg game. Comparable to a mixed equilibrium strategy, (Bulò, Biggio, Pillai, Pelillo, & Roli, 2017) studied randomized defense strategy based on a game-theoretic formulation.

3.2 Privacy Preserving Machine Learning

Privacy preserving mechanisms are developed to prevent privacy leakage directly from a database or through a learning model. There are two main privacy preserving mechanisms, k -anonymity (Samarati & Sweeney, 1998; Samarati, 2001; Sweeney, 2002) and differential privacy (Dwork & Smith, 2010a; Dwork, Kenthapadi, McSherry, Mironov, & Naor, 2006; Dwork, McSherry, Nissim, & Smith, 2006). k -anonymity directly perturbs the individual data points, whereas differential privacy injects noises to query results from a database. Two privacy preserving mechanisms can be considered as complimentary to each other, as suggested in (Clifton & Tassa, 2013). (Aggarwal & Philip, 2008) provided a survey of k -anonymity and distributed privacy preserving techniques and their application to data mining tasks. (Fung, Wang, Chen, & Yu, 2010) focused on various privacy models for publishing privacy protected data records, not the results from data mining or machine learning algorithms. Both k -anonymity and differential privacy were discussed among other approaches. Furthermore (Dwork & Smith, 2010b) focused on applying differential privacy to statistics, such as statistical inference and robust statistics. Below is a brief summary of k -anonymity and differential privacy.

- A database satisfies k -anonymity if every unique tuple for every combination of the variables appears at least k times.
- Differential privacy is a more sophisticated mechanism. Let Q be a randomized function used to release information from a database. Q satisfies ϵ -differential privacy if for

two databases D and D' differing by one record,

$$\frac{\Pr(Q(D) \in H)}{\Pr(Q(D') \in H)} \leq e^\epsilon, \quad \forall H \in \text{range}(Q).$$

The Laplace mechanism adds a Laplace noise to a query function q to generate the randomized query result satisfying ϵ - differential privacy. An important concept to determine the size of the Laplace noise is the sensitivity of the query function q . q 's sensitivity equals to the maximum change of q values over two databases differing by one record, D and D' . For Laplace mechanism, the Laplace noise added query output is

$$q(D) + \text{Laplace} \left(\frac{\text{sensitivity}}{\epsilon} \right),$$

where ϵ is a pre-determined parameter (and hence the name ϵ - differential privacy). The recommended values for ϵ vary in a big interval, from as small as 0.01 and 0.1 to as big as 7, e.g., (J. Lee & Clifton, 2011).

Differential privacy has been used to provide privacy guarantee to machine learning models. For example, (Friedman & Schuster, 2010; Jagannathan, Pillaipakkamnatt, & Wright, 2009) produced privacy preserving decision trees; (Rubinstein, Bartlett, Huang, & Taft, 2012) produced privacy preserving SVM; (Abadi et al., 2016; Shokri & Shmatikov, 2015) introduced privacy guarantee to deep learning. However, a recent membership inference attack against differentially private deep learning model (Rahman, Rahman, Laganier, Mohammed, & Wang, 2018) suggested more research needs to be done to prevent privacy leakage from complex learning models such as DNNs. There are several efforts along this line. (Nasr, Shokri, & Houmansadr, 2018) developed a min-max game and an adversarial training procedure to protect DNNs from membership inference attacks. Similarly (Hayes & Ohrimenko, 2018) also developed an adversarial training procedure to defend against membership inference attacks when multiple parties shared their data. (Veale, Binns, & Edwards, 2018) mentioned the possibility of extending data protection law to machine learning models by considering them as personal data in legal terms, which pointed to an ultimate solution through regulation process.

3.3 Defense Against Poisoning Attacks

To remove poisonous attack instances from the training data is a difficult task. Here we discuss all the existing approaches proposed to defend against poisoning attacks. We need more systematic approaches to evaluate the impact of poisoning attacks and mitigate the threat on learning techniques.

- (Nelson et al., 2008) proposed Reject On Negative Impact (RONI) defense and dynamic threshold defense. The idea behind RONI was similar to that for outlier detection in linear regression. The impact of every training email was measured as the difference in performance by including and excluding it in the training process. The training emails with large negative impact were removed from the training data. However RONI did not perform well in targeted poisoning attacks. Meanwhile, dynamic threshold defense recommended dynamically adjusting threshold values in SpamBayes. Although this approach increased accuracy with legitimate emails, it had difficulty to correctly label spam emails.
- Facing poisoning attacks against principal component analysis (PCA) subspace anomaly detection method in backbone network, (Rubinstein et al., 2009) proposed a robust PCA based defense approach. It maximized Median Absolute Deviation (MAD) instead of variance to compute principal components, and used a robust threshold value based on Laplace distribution instead of Gaussian. (Madani & Vlajic, 2018) built an autoencoder based intrusion detection system, assuming malicious poisoning attack instances were under 2%. It demonstrated that the autoencoder was more robust compared with a PCA based one.
- (Cretu, Stavrou, Locasto, Stolfo, & Keromytis, 2008) proposed a training data sanitization scheme. The training data was broken into several subsets, where subsets that belonged to different networks or domains yielded good results in the sanitization process. Each subset was used to train a model. These models were used to label every training data point. A voting scheme was used to determine whether a training data point was an attack instance or not. Potential attack instances were then removed from training data.

- (Biggio, Corona, Fumera, Giacinto, & Roli, 2011) argued poisoning attack instances can be viewed as a special type of outliers. Because bagging can reduce the impact of outliers in training, bagging ensembles constructed for spam filtering and intrusion detection offered promising results.
- (Steinhardt, Koh, & Liang, 2017) analyzed the upper bounds on the loss caused by poisoning attacks against SVM. Outliers were defined as points far from cluster centroids, which were computed with or without the poisonous points. An empirical online learning algorithm was developed to compute the upper bound on the worst case loss for any given dataset.
- (B. Chen et al., 2019) proposed activation clustering method to detect the backdoor trigger in the training data. The method examined activation of the last hidden layer in a DNN, where the poisonous samples containing the backdoor trigger appeared in a separate cluster.

3.4 Robust DNN

Unfortunately robustifying DNN proves to be a much more difficult task compared to finding more effective adversarial samples. Defense strategies were quickly found to be vulnerable to newer and stronger attacks. Here we discuss the main defense strategies proposed in the literature.

- **Adversarial Training:** The main idea behind adversarial training is to further regularize a DNN and improve its robustness. The procedure includes adversarial samples in training, and continuously generating new adversarial samples at every step of training (Szegedy et al., 2014; I. J. Goodfellow et al., 2015; Kurakin et al., 2017b; Tramèr et al., 2018). (Kurakin et al., 2017b) reported “label leaking” for the adversarially trained network – it is more accurate with the adversarial images generated using the same attack algorithm than the clean images. An improved procedure, ensemble adversarial training, used adversarial samples generated against other pre-trained models to decouple the attack algorithm and the model under attack. (Zantedeschi, Nicolae,

& Rawat, 2017) proposed to enlarge the training data by adding Gaussian noise to clean images, which is computationally less expensive than using adversarial samples.

- **Distillation:** (Papernot, McDaniel, Wu, et al., 2016) proposed defensive distillation using two networks. The first network was trained on the original training data. The second network was trained on data with the soft labels produced from the first network instead of the original labels. Meanwhile, (Wang et al., 2018) showed experimentally Bayesian neural networks (BNN) and distilled BNN accurately detected adversarial samples if adversarial perturbations were generated with respect to only one posterior sample of the network weights.
- **Gradient Masking:** Gradient masking is another popular approach to defend against adversarial samples, e.g., (Lyu, Huang, & Liang, 2015; Shaham, Yamada, & Negahban, 2018; Nguyen, Wang, & Sinha, 2018; Ross & Doshi-Velez, 2018; Li & Li, 2017). There are three main gradient masking approaches: 1) A defense can cause gradient shattering, where gradients do not exist or point to the wrong direction, to render gradient based attacks ineffective; 2) A randomized defense can cause the gradients to be randomized as well, and reduce the effectiveness of attacks; 3) A defense can cause vanishing or exploding gradients when it involves several iterations of network computations.
- **Pre-processing:** Carefully designed pre-processing procedures were also proposed to mitigate the effect of adversarial perturbations. (Xu, Evans, & Qi, 2018) proposed feature squeezing via spatial smoothing or reducing pixel color bit depth. (Guo, Rana, Cisse, & van der Maaten, 2018) discovered image transformations, such as total variance minimization and image quilting, helped to eliminate adversarial perturbations. (Samangouei, Kabkab, & Chellappa, 2018) proposed to use a GAN to denoise adversarial samples before feeding them into a classifier. Meanwhile, several winning teams in the 2017 NIPS competition (Kurakin et al., 2018) applied image denoising techniques to remove adversarial perturbations.

- **Detection:** There are many detection approaches proposed. For example, (Metzen, Genewein, Fischer, & Bischoff, 2017) proposed using a subnetwork as a detector. (K. Lee, Lee, Lee, & Shin, 2018) computed a confidence score to detect both adversarial samples and out of class samples. (Song, Kim, Nowozin, Ermon, & Kushman, 2018) used statistical hypothesis tests to detect adversarial samples that were different from the clean image distributions.

At the same time, many of these approaches were showed to be ineffective. (Carlini & Wagner, 2017b) developed the C&W attack specifically to break distillation defense. (Carlini & Wagner, 2017a) showed 10 detection methods failed the C&W attack soon after they were published. (Athalye, Carlini, & Wagner, 2018) showed gradient masking was still vulnerable to attacks. (I. J. Goodfellow et al., 2015) had an experiment to demonstrate that an ensemble was not more robust against adversarial perturbations. (Tramèr et al., 2018) mentioned adversarial training was vulnerable to black box attacks. Since a robust learning model must withstand different types of attacks, to build a robust DNN remains a major challenge as of today.

4 CONCLUSIONS

Generative Adversarial Networks (GAN) (e.g., (I. Goodfellow et al., 2014; Zhao, Mathieu, & LeCun, 2017; Arjovsky, Chintala, & Bottou, 2017)) is sometimes mentioned together with adversarial machine learning. GAN is a system deploying two competing neural networks, i.e., a generator and a discriminator, to produce ultra-high dimensional samples such as images. GAN is used also for image super-resolution (e.g., (Ledig et al., 2017)), for generating 3D objects (J. Wu, Zhang, Xue, Freeman, & Tenenbaum, 2016), for generating new molecules (Zhavoronkov et al., 2019) etc. GAN is essentially a generative model. It is an important research topic but follows a different direction compared with adversarial machine learning, which focuses on learning model vulnerabilities and robustness. For the interested readers, a GitHub site ⁴ has a list of GAN papers.

⁴<https://github.com/nightrome/really-awesome-gan> (last verified on 03/01/2020)

Most of the existing research in adversarial machine learning focuses on supervised learning. On the other hand, providing labels for a large amount of data points or samples from the newest attacks may require expensive human expertise and becomes a significant bottleneck. How to identify adversarial samples in unsupervised and weakly supervised scenarios needs to receive more attention. Clustering techniques and active learning techniques also need to be robustified facing adversaries, e.g., (Bayer, Comparetti, Hlauschek, Kruegel, & Kirda, 2009; Lin, Ke, & Tsai, 2015; Pi, Lu, Sagduyu, & Chen, 2016; Miller, Hu, Qiu, & Kesidis, 2017; Zhou, Kantarcioglu, & Xi, 2019). Meanwhile, it is important to quantify the robustness and accuracy trade-off for machine learning algorithms facing adversarial attacks. Although recent works started to propose certain robustness or uncertainty measures, a more in-depth study on the trade-off is needed to build resilient learning algorithms.

Compared with a large number of digital attacks, there are only a handful of physical attacks, e.g., (Sharif, Bhagavatula, Bauer, & Reiter, 2016; Eykholt et al., 2018; Athalye, Engstrom, Ilyas, & Kwok, 2018; Kurakin et al., 2017a; Zhang et al., 2017; Xi, Chen, Fei, Tu, & Deng, 2020). They pose a more dangerous threat for learning techniques. Physical attacks either design physical objects that cannot be recognized by learning models (Eykholt et al., 2018; Athalye, Engstrom, et al., 2018; Kurakin et al., 2017a), or they create malicious data targeting a particular vulnerability in a hardware device (Zhang et al., 2017). (Xi et al., 2020) designed a moving physical object to make an object detection system “blind” because of the motion of the object. A successful defense against physical attacks may require multiple sensor systems to work together with robust learning techniques and more secure hardware designs.

We cannot stop adversaries from launching unexpected new attacks. However our efforts on secure and robust machine learning techniques can mitigate the damages caused by adversarial objects and slow down the arms race between adversaries and defenders.

ACKNOWLEDGMENTS

The author would like to thank the reviewers for their helpful comments, which greatly improved this article.

FUNDING INFORMATION

This work is supported in part by ARO grant W911NF-17-1-0356, Purdue CERIAS seed grant, and a grant by Northrop Grumman Corp.

CONFLICT OF INTEREST

The author has declared no conflicts of interest for this article.

References

- Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. In *Proceedings of the 2016 acm sigsac conference on computer and communications security* (pp. 308–318).
- Aggarwal, C. C., & Philip, S. Y. (2008). A general survey of privacy-preserving data mining models and algorithms. In *Privacy-preserving data mining* (pp. 11–52). Springer.
- Akhtar, N., & Mian, A. (2018). Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, *6*, 14410–14430.
- Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th international conference on machine learning (icml)* (Vol. 70, pp. 214–223).
- Athalye, A., Carlini, N., & Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning* (pp. 274–283).
- Athalye, A., Engstrom, L., Ilyas, A., & Kwok, K. (2018, 10–15 Jul). Synthesizing robust adversarial examples. In *Proceedings of the 35th international conference on machine learning* (pp. 284–293). Stockholmsmssan, Stockholm Sweden.
- Baluja, S., & Fischer, I. (2018). Learning to attack: Adversarial transformation networks. In *Thirty-second aaii conference on artificial intelligence* (pp. 1–9).
- Barreno, M., Nelson, B., Joseph, A. D., & Tygar, J. D. (2010). The security of machine learning. *Machine Learning*, *81*(2), 121–148.

- Bayer, U., Comparetti, P. M., Hlauschek, C., Kruegel, C., & Kirda, E. (2009). Scalable, behavior-based malware clustering. In *Ndss* (Vol. 9, pp. 8–11).
- Biggio, B., Corona, I., Fumera, G., Giacinto, G., & Roli, F. (2011). Bagging classifiers for fighting poisoning attacks in adversarial classification tasks. In *International workshop on multiple classifier systems* (pp. 350–359).
- Biggio, B., Fumera, G., & Roli, F. (2014). Security evaluation of pattern classifiers under attack. *IEEE transactions on knowledge and data engineering*, 26(4), 984–996.
- Biggio, B., Nelson, B., & Laskov, P. (2012). Poisoning attacks against support vector machines. In *29th international conference on machine learning* (pp. 1807–1814).
- Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84, 317–331.
- Brückner, M., & Scheffer, T. (2011). Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th acm sigkdd international conference on knowledge discovery and data mining* (pp. 547–555).
- Bulò, S. R., Biggio, B., Pillai, I., Pelillo, M., & Roli, F. (2017). Randomized prediction games for adversarial machine learning. *IEEE transactions on neural networks and learning systems*, 28(11), 2466–2478.
- Carlini, N., & Wagner, D. (2017a). Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th acm workshop on artificial intelligence and security* (pp. 3–14).
- Carlini, N., & Wagner, D. (2017b). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)* (pp. 39–57).
- Chen, B., Carvalho, W., Baracaldo, N., Ludwig, H., Edwards, B., Lee, T., . . . Srivastava, B. (2019). Detecting backdoor attacks on deep neural networks by activation clustering. In *The AAAI’s workshop on artificial intelligence safety (safeai 2019)* (pp. 1–8).
- Chen, P.-Y., Sharma, Y., Zhang, H., Yi, J., & Hsieh, C.-J. (2018). Ead: elastic-net attacks to deep neural networks via adversarial examples. In *Thirty-second AAAI conference on artificial intelligence* (pp. 1–9).
- Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., & Hsieh, C.-J. (2017). Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute

- models. In *Proceedings of the 10th acm workshop on artificial intelligence and security* (pp. 15–26).
- Cleverhans adversarial examples library*. (n.d.). <https://github.com/tensorflow/cleverhans>.
- Clifton, C., & Tassa, T. (2013). On syntactic anonymity and differential privacy. In *Proceedings of the ieee 29th international conference on data engineering workshops (icdew)* (pp. 88–93).
- Corona, I., Giacinto, G., & Roli, F. (2013). Adversarial attacks against intrusion detection systems: Taxonomy, solutions and open issues. *Information Sciences*, *239*, 201–225.
- Cranor, L. F., & LaMacchia, B. A. (1998). Spam! *Communications of the ACM*, *41*(8), 74–83.
- Cretu, G. F., Stavrou, A., Locasto, M. E., Stolfo, S. J., & Keromytis, A. D. (2008). Casting out demons: Sanitizing training data for anomaly sensors. In *Ieee symposium on security and privacy* (pp. 81–95).
- Dalvi, N., Domingos, P., Mausam, Sanghai, S., & Verma, D. (2004). Adversarial classification. In *Proceedings of the tenth acm sigkdd international conference on knowledge discovery and data mining* (pp. 99–108).
- Dekel, O., Shamir, O., & Xiao, L. (2010). Learning to classify with missing and corrupted features. *Machine Learning*, *81*, 149–178.
- Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., & Li, J. (2018). Boosting adversarial attacks with momentum. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 9185–9193).
- Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., & Naor, M. (2006). Our data, ourselves: Privacy via distributed noise generation. In *Annual international conference on the theory and applications of cryptographic techniques* (pp. 486–503).
- Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference* (pp. 265–284).
- Dwork, C., & Smith, A. (2010a). Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, *1*(2), 135–154.
- Dwork, C., & Smith, A. (2010b). Differential privacy for statistics: What we know and what

- we want to learn. *Journal of Privacy and Confidentiality*, 1(2), 135–154.
- Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., . . . Song, D. (2018, June). Robust physical-world attacks on deep learning visual classification. In *The IEEE conference on computer vision and pattern recognition (cvpr)* (pp. 1625–1634).
- Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security* (pp. 1322–1333).
- Friedman, A., & Schuster, A. (2010). Data mining with differential privacy. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 493–502).
- Fung, B. C., Wang, K., Chen, R., & Yu, P. S. (2010). Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys*, 42(4), 1–53.
- Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)* (pp. 1050–1059).
- Globerson, A., & Roweis, S. (2006). Nightmare at test time: robust learning by feature deletion. In *Proceedings of the 23rd international conference on machine learning* (pp. 353–360).
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).
- Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *International conference on learning representations* (pp. 1–12).
- Grosse, K., Papernot, N., Manoharan, P., Backes, M., & McDaniel, P. (2017). Adversarial examples for malware detection. In *European symposium on research in computer security* (pp. 62–79).
- Gu, T., Dolan-Gavitt, B., & Garg, S. (2017). Badnets: Identifying vulnerabilities in the machine learning model supply chain. In *NIPS machine learning and computer security workshop* (pp. 1–5).
- Guo, C., Rana, M., Cisse, M., & van der Maaten, L. (2018). Countering adversarial images

- using input transformations. In *Proceedings of the international conference on learning representations (iclr)* (pp. 1–10).
- Hayes, J., & Ohrimenko, O. (2018). Contamination attacks and mitigation in multi-party machine learning. In *Advances in neural information processing systems* (pp. 6604–6616).
- Householder, A., Houle, K., & Dougherty, C. (2002). Computer attack trends challenge internet security. *Computer*, *35*(4), sulp5–sulp7.
- Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I., & Tygar, J. (2011). Adversarial machine learning. In *Proceedings of the 4th acm workshop on security and artificial intelligence* (pp. 43–58).
- IBM Adversarial Robustness Toolbox (ART v0.8.0). (n.d.).
<https://github.com/IBM/adversarial-robustness-toolbox>.
- Ilyas, A., Engstrom, L., Athalye, A., & Lin, J. (2018). Black-box adversarial attacks with limited queries and information. In *Proceedings of the 6th international conference on learning representations (iclr)* (pp. 1–10).
- Jagannathan, G., Pillaipakkamnatt, K., & Wright, R. N. (2009). A practical differentially private random decision tree classifier. In *Proceedings of ieee international conference on data mining workshops* (pp. 114–121).
- Joseph, A. D., Laskov, P., Roli, F., Tygar, J. D., & Nelson, B. (2013). Machine learning methods for computer security (dagstuhl perspectives workshop 12371). In *Dagstuhl manifestos* (Vol. 3).
- Kantarcioğlu, M., Xi, B., & Clifton, C. (2011). Classifier evaluation and attribute selection against active adversaries. *Data Mining and Knowledge Discovery*, *22*(1-2), 291–335.
- Kearns, M. J., & Li, M. (1993). Learning in the presence of malicious errors. *SIAM Journal on Computing*, *22*(4), 807-837.
- Kloft, M., & Laskov, P. (2010). Online anomaly detection under adversarial impact. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 405–412).
- Kurakin, A., Goodfellow, I., & Bengio, S. (2017a). Adversarial examples in the physical world. In *Proceedings of the 6th international conference on learning representations*

- (*iclr*) (pp. 1–10).
- Kurakin, A., Goodfellow, I., & Bengio, S. (2017b). Adversarial machine learning at scale. In *Proceedings of the 6th international conference on learning representations (iclr)* (pp. 1–10).
- Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., ... others (2018). Adversarial attacks and defences competition. In *The nips'17 competition: Building intelligent systems* (pp. 195–231). Springer.
- Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems (NIPS)* (pp. 6402–6413).
- Lanckriet, G. R. G., Ghaoui, L. E., Bhattacharyya, C., & I., J. M. (2002). A robust minimax approach to classification. *Journal of Machine Learning Research*, 3, 555–582.
- Laskov, P., & Kloft, M. (2009). A framework for quantitative security analysis of machine learning. In *Proceedings of the 2nd acm workshop on security and artificial intelligence* (pp. 1–4).
- Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., ... others (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4681–4690).
- Lee, J., & Clifton, C. (2011). How much is enough? choosing ϵ for differential privacy. In *International conference on information security* (pp. 325–340).
- Lee, K., Lee, K., Lee, H., & Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in neural information processing systems* (pp. 7167–7177).
- Lee, W., Stolfo, S. J., et al. (1998). Data mining approaches for intrusion detection. In *Usenix security symposium* (pp. 79–93).
- Li, X., & Li, F. (2017). Adversarial examples detection in deep networks with convolutional filter statistics. In *Proceedings of the ieee international conference on computer vision* (pp. 5764–5772).
- Lin, W.-C., Ke, S.-W., & Tsai, C.-F. (2015). Cann: An intrusion detection system based on

- combining cluster centers and nearest neighbors. *Knowledge-based systems*, 78, 13–21.
- Liu, W., & Chawla, S. (2010). Mining adversarial patterns via regularized loss minimization. *Machine learning*, 81(1), 69–83.
- Liu, X., Yang, H., Liu, Z., Song, L., Li, H., & Chen, Y. (2019). Dpatch: An adversarial patch attack on object detectors. In *The aaai’s workshop on artificial intelligence safety (safeai 2019)* (pp. 1–8).
- Liu, Y., Chen, X., Liu, C., & Song, D. (2017). Delving into transferable adversarial examples and black-box attacks. In *Proceedings of the 6th international conference on learning representations (iclr)* (pp. 1–10).
- Lowd, D., & Meek, C. (2005). Good word attacks on statistical spam filters. In *2nd conf. email and anti-spam (ceas)* (pp. 1–8).
- Lyu, C., Huang, K., & Liang, H.-N. (2015). A unified gradient regularization family for adversarial examples. In *2015 ieee international conference on data mining* (pp. 301–309).
- Madani, P., & Vlazic, N. (2018). Robustness of deep autoencoder in intrusion detection under adversarial contamination. In *Proceedings of the 5th annual symposium and bootcamp on hot topics in the science of security* (p. 1).
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *Proceedings of the 6th international conference on learning representations (iclr)* (pp. 1–10).
- Malinin, A., & Gales, M. (2018). Predictive uncertainty estimation via prior networks. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 7047–7058).
- Melis, L., Song, C., De Cristofaro, E., & Shmatikov, V. (2019). Exploiting unintended feature leakage in collaborative learning. In *2019 ieee symposium on security and privacy* (pp. 1–16).
- Metzen, J. H., Genewein, T., Fischer, V., & Bischoff, B. (2017). On detecting adversarial perturbations. In *Proceedings of the international conference on learning representations (iclr)* (pp. 1–10).
- Miller, D. J., Hu, X., Qiu, Z., & Kesidis, G. (2017). Adversarial learning: a critical review and active learning study. In *2017 ieee 27th international workshop on machine learning*

- for signal processing* (pp. 1–6).
- Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1765–1773).
- Moosavi-Dezfooli, S.-M., Fawzi, A., & Frossard, P. (2016). Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2574–2582).
- Mukkamala, S., Janoski, G., & Sung, A. (2002). Intrusion detection using neural networks and support vector machines. In *Neural networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on* (Vol. 2, pp. 1702–1707).
- Nasr, M., Shokri, R., & Houmansadr, A. (2018). Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security* (pp. 634–646).
- Nelson, B., Barreno, M., Chi, F. J., Joseph, A. D., Rubinstein, B. I., Saini, U., . . . Xia, K. (2008). Exploiting machine learning to subvert your spam filter. In *Proceedings of the 1st usenix workshop on large-scale exploits and emergent threats* (pp. 7:1–7:9).
- Nguyen, L., Wang, S., & Sinha, A. (2018). A learning and masking approach to secure learning. In *International conference on decision and game theory for security* (pp. 453–464).
- Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroSec)* (pp. 372–387).
- Papernot, N., McDaniel, P., Wu, X., Jha, S., & Swami, A. (2016). Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy* (pp. 582–597).
- Papernot, N., McDaniel, P. D., Goodfellow, I. J., Jha, S., Celik, Z. B., & Swami, A. (2017). Practical black-box attacks against deep learning systems using adversarial examples. In *Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security* (p. 506–519).
- Pi, L., Lu, Z., Sagduyu, Y., & Chen, S. (2016). Defending active learning against adversarial

- inputs in automated document classification. In *2016 IEEE Global Conference on Signal and Information Processing* (pp. 257–261).
- Rahman, M. A., Rahman, T., Laganiere, R., Mohammed, N., & Wang, Y. (2018). Membership inference attack against differentially private deep learning model. *Transactions on Data Privacy*, 11(1), 61–79.
- Ross, A. S., & Doshi-Velez, F. (2018). Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-second AAAI conference on artificial intelligence*.
- Rozsa, A., Rudd, E. M., & Boulton, T. E. (2016). Adversarial diversity and hard positive generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 25–32).
- Rubinfeld, B. I., Bartlett, P. L., Huang, L., & Taft, N. (2012). Learning in a large function space: Privacy-preserving mechanisms for SVM learning. *Journal of Privacy and Confidentiality*, 4(1), 65–100.
- Rubinfeld, B. I., Nelson, B., Huang, L., Joseph, A. D., Lau, S.-h., Rao, S., . . . Tygar, J. D. (2009). Antidote: understanding and defending against poisoning of anomaly detectors. In *Proceedings of the 9th ACM SIGCOMM conference on internet measurement* (pp. 1–14).
- Ryan, J., Lin, M.-J., & Miikkulainen, R. (1998). Intrusion detection with neural networks. In *Advances in neural information processing systems* (pp. 943–949).
- Samangouei, P., Kabkab, M., & Chellappa, R. (2018). Defense-gan: Protecting classifiers against adversarial attacks using generative models. In *the sixth international conference on learning representations (ICLR 2018)*.
- Samarati, P. (2001). Protecting respondents identities in microdata release. *IEEE transactions on Knowledge and Data Engineering*, 13(6), 1010–1027.
- Samarati, P., & Sweeney, L. (1998). Generalizing data to provide anonymity when disclosing information. In *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART symposium on principles of database systems (PODS)* (Vol. 98, p. 188).
- Servedio, R. A. (2001). Smooth boosting and learning with malicious noise. In *Fourteenth annual conference on computational learning theory (COLT)* (pp. 473–489).

- Shafahi, A., Huang, W. R., Studer, C., Feizi, S., & Goldstein, T. (2019). Are adversarial examples inevitable? In *Proceedings of the international conference on learning representations (iclr)* (pp. 1–10).
- Shaham, U., Yamada, Y., & Negahban, S. (2018). Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, *307*, 195–204.
- Sharif, M., Bhagavatula, S., Bauer, L., & Reiter, M. K. (2016). Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security* (pp. 1528–1540).
- Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. In *Proceedings of the 22nd acm sigsac conference on computer and communications security* (pp. 1310–1321).
- Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy* (pp. 3–18).
- Song, Y., Kim, T., Nowozin, S., Ermon, S., & Kushman, N. (2018). Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *Proceedings of the international conference on learning representations (iclr)* (pp. 1–10).
- Šrndić, N., & Laskov, P. (2013). Detection of malicious pdf files based on hierarchical document structure. In *Proceedings of the 20th annual network & distributed system security symposium* (pp. 1–16).
- Steinhardt, J., Koh, P. W. W., & Liang, P. S. (2017). Certified defenses for data poisoning attacks. In *Advances in neural information processing systems* (pp. 3517–3529).
- Su, J., Vargas, D. V., & Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*.
- Sung, A. H., & Mukkamala, S. (2003). Identifying important features for intrusion detection using support vector machines and neural networks. In *Applications and the internet, 2003. proceedings. 2003 symposium on* (pp. 209–216).
- Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557–570.

- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. In *Proceedings of the international conference on learning representations (iclr)* (pp. 1–10).
- Teo, C. H., Globerson, A., Roweis, S. T., & Smola, A. J. (2008). Convex learning with invariances. In *Advances in neural information processing systems* (pp. 1489–1496).
- Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. In *International conference on learning representations* (pp. 1–12).
- Tsai, C.-F., Hsu, Y.-F., Lin, C.-Y., & Lin, W.-Y. (2009). Intrusion detection by machine learning: A review. *Expert Systems with Applications*, *36*(10), 11994–12000.
- Valiant, L. G. (1985). Learning disjunctions of conjunctions. In *Proceedings of the 9th international joint conference on artificial intelligence* (p. 560-566).
- Veale, M., Binns, R., & Edwards, L. (2018). Algorithms that remember: model inversion attacks and data protection law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *376*(2133), 20180083.
- Wang, K.-C., Vicol, P., Lucas, J., Gu, L., Grosse, R., & Zemel, R. (2018). Adversarial distillation of bayesian neural network posteriors. In *International conference on machine learning* (pp. 5177–5186).
- Weng, T.-W., Zhang, H., Chen, P.-Y., Yi, J., Su, D., Gao, Y., ... Daniel, L. (2018). Evaluating the robustness of neural networks: An extreme value theory approach. , 1–10.
- Wu, J., Zhang, C., Xue, T., Freeman, B., & Tenenbaum, J. (2016). Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in neural information processing systems* (pp. 82–90).
- Wu, X., Fredrikson, M., Jha, S., & Naughton, J. F. (2016). A methodology for formalizing model-inversion attacks. In *2016 IEEE 29th computer security foundations symposium* (pp. 355–370).
- Xi, B., Chen, Y., Fei, F., Tu, Z., & Deng, X. (2020). Bio-inspired adversarial attack against deep neural networks. In *Proceedings of the workshop on artificial intelligence safety (safeai 2020) co-located with 34th aai conference on artificial intelligence (aaai 2020)*

- (pp. 1–5).
- Xiao, H., Biggio, B., Brown, G., Fumera, G., Eckert, C., & Roli, F. (2015). Is feature selection secure against training data poisoning? In *International conference on machine learning* (pp. 1689–1698).
- Xu, W., Evans, D., & Qi, Y. (2018). Feature squeezing: Detecting adversarial examples in deep neural networks. In *Proceedings of network and distributed systems security symposium (ndss)* (pp. 1–15).
- Yuan, X., He, P., Zhu, Q., & Li, X. (2019). Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 1–20.
- Zantedeschi, V., Nicolae, M.-I., & Rawat, A. (2017). Efficient defenses against adversarial attacks. In *Proceedings of the 10th acm workshop on artificial intelligence and security* (pp. 39–49).
- Zhang, G., Yan, C., Ji, X., Zhang, T., Zhang, T., & Xu, W. (2017). Dolphinattack: Inaudible voice commands. In *Proceedings of the 2017 acm sigsac conference on computer and communications security* (pp. 103–117).
- Zhao, J., Mathieu, M., & LeCun, Y. (2017). Energy-based generative adversarial networks. In *5th international conference on learning representations (iclr)* (pp. 1–17).
- Zhavoronkov, A., Ivanenkov, Y. A., Aliper, A., Veselov, M. S., Aladinskiy, V. A., Aladin-skaya, A. V., ... others (2019). Deep learning enables rapid identification of potent ddr1 kinase inhibitors. *Nature biotechnology*, 37(9), 1038–1040.
- Zhou, Y., & Kantarcioglu, M. (2016). Modeling adversarial learning as nested stackelberg games. In *Proceedings of the pacific-asia conference on knowledge discovery and data mining* (pp. 350–362).
- Zhou, Y., Kantarcioglu, M., & Xi, B. (2019). Adversarial active learning. In *The pacific-asia conference on knowledge discovery and data mining (pakdd) workshop on weakly supervised learning: Progress and future* (pp. 1–12).