A Taxonomy and Survey of Attacks Against Machine Learning

Nikolaos Pitropakis[a], Emmanouil Panaousis[b], Thanassis Giannetsos[c], Eleftherios Anastasiadis[d], George Loukas[e]

[a] *Edinburgh Napier University, UK*
[b] *University of Surrey, UK*
[c] *Technical University of Denmark, Denmark*
[d] *Imperial College London, UK*
[e] *University of Greenwich, UK*

Abstract

The majority of machine learning methodologies operate with the assumption that their environment is benign. However, this assumption does not always hold, as it is often advantageous to adversaries to maliciously modify the training (poisoning attacks) or test data (evasion attacks). Such attacks can be catastrophic given the growth and the penetration of machine learning applications in society. Therefore, there is a need to secure machine learning enabling the safe adoption of it in adversarial cases, such as spam filtering, malware detection, and biometric recognition. This paper presents a taxonomy and survey of attacks against systems that use machine learning. It organizes the body of knowledge in adversarial machine learning so as to identify the aspects where researchers from different fields can contribute to. The taxonomy identifies attacks which share key characteristics and as such can potentially be addressed by the same defense approaches. Thus, the proposed taxonomy makes it easier to understand the existing attack landscape towards developing defence mechanisms, which are not investigated in this survey. The taxonomy is also leveraged to identify open problems that can lead to new research areas within the field of adversarial machine learning.

Keywords: Machine learning, attacks, taxonomy, survey

1. Introduction

In recent years, tremendous progress has been made in Machine Learning (ML) and its use has become ubiquitous in many emerging applications where data can be collected and processed locally, at edge or the cloud. This data can be used for training of machine learning models, which in turn can be deployed for example to perform predictions or support decision making in healthcare [1], intrusion detection [2], fraud detection [3], autonomous vehicles [4] and many other applications [5–7].

This rapidly expanding adoption of ML technologies, however, has rendered them attractive targets to adversaries who want to manipulate such mechanisms for malevolent purposes [8]. All ML systems are trained using datasets that are assumed to be *representative* and *trustworthy* for the subject matter in question thus enabling the construction of a valid system perception of the phenomenon of interest. However, malicious actors can impact the decision-making algorithms of such approaches by either targeting the training data or forcing the model to their desired output, e.g., misclassification of abnormal events. These types of attacks, known as *poisoning* and *evasion* attacks [9] respectively, allow adversaries to significantly decrease overall performance, cause targeted misclassification or bad behaviour, and insert backdoors and neural Trojans [8, 10].

Adversarial Machine Learning (AML) sits at the intersection of machine learning and cyber security, and it is often defined as the study of effective machine learning techniques against an adversarial opponent [10]. For example, Huang et al. [11] propose a new method that learns robust classifiers from supervised data by generating adversarial examples as an intermediate step facilitating the attack detection. Representative examples of applications that AML can be applied to include intrusion detection, spam filtering, visual recognition and biometrics authentication.

What is missing is an in-depth investigation of all the possible AML attack vectors (i.e., their configuration, execution, strategy, impact, etc.) that can lead to a detailed taxonomy of attacks and threats against the various phases of the

ML pipeline. This can enable the design and implementation of more appropriate mitigation countermeasures. Currently, only a small number of related taxonomies and surveys have been published. Adversarial behaviour has also been investigated as part of online learning Markov Decision Processes (MDPs) when both the transition distributions and loss functions are chosen by an adversary [12, 13].

1.1. Motivation & Contributions

Motivated by these issues, the aim of this investigation is to provide a complete analysis and taxonomy of the types of malicious attacks against the entire ML pipeline. In this context, we address the natural limitations of the early taxonomies by providing a new, comprehensive categorization, covering both old (pre-2011), newer (post-2011) and potential areas of research that may arise in the future in a complete landscape of adversarial machine learning. We use this taxonomy as the basis for surveying and classifying the different available approaches, especially emphasizing on the ones presented over the last two years, and identifying which areas could benefit considerably from further research. In summary, this paper's contributions can be summarized as follows:

(*i*) a comprehensive taxonomy of the characteristics of AML approaches (Section 2),

(*ii*) a systematic review of the landscape of existing AML approaches towards their categorization and classification, following the proposed taxonomy, for the aforementioned application domains (Section 3) and,

(*iii*) identification of open issues and specific areas where further research can have considerable impact (Section 4).

Through this investigation, our main goal is to shed light on this emerging attack paradigm so that it can be used as a baseline for the design of more robust countermeasures; so that machine learning can offer enhanced security and privacy capabilities that can further accelerate data-driven insights and knowledge acquisition.

1.2. Related Work

From a cyber security perspective, there are multiple avenues for adversely affecting a system that is based on machine learning for example by targeting its data collection or transmission, and manipulating its models developed or their input (Figure 1). Remarkably, for such a research area that has extensive societal impact, only a small number of related taxonomies and surveys have been published. In 2011, Huang et al. [10] published an early taxonomy of adversarial machine learning including categories for influence, security violation and specificity. This has served as an excellent basis for attack categorization, but over the years it has naturally become less able to cover newer approaches.

Adversarial attacks against machine learning have become a reality in a variety of applications. *Intrusion detection* is a very important defence mechanism, which can be bypassed by adversaries who achieve to increase false negatives or both false positives and false negatives. In addition, the importance of *spam filtering*, given the growth of attacks against email classification, is prominent making it another key use case for researchers. Furthermore, the growth of deep neural networks and the proliferation of automated vehicles has made *visual recognition* an attractive field for adversaries to attack machine learning (e.g., [14–20]).

Recently Shumailov et al. [21] published a work where they use taboo behaviours both subtle and diverse to detect adversarial attacks and make a classifier more robust against them. Most papers surveyed investigate attacks within this domain and most of them are concerned with perturbed samples that confuse the machine learning systems. Nevertheless, adversarial attacks have been noticed in a variety of other applications (or combination of them) such as recommendation systems, autoregressive forecasting models, biometric recognition systems, credit card fraud detection systems. There are also attacks aiming at different applications of machine learning, or attacks which were initially created

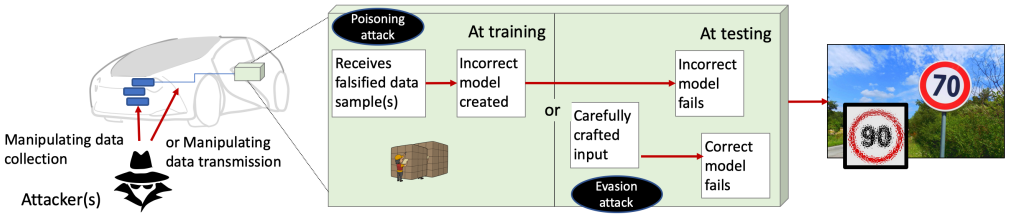

Figure 1: Different avenues for attacking a machine learning based system

for one application and might be equally effective in a variety of others.

This is not the first survey in the field of adversarial machine learning but there are distinguished differences between the existing literature and our paper. The proposed taxonomy, not only offers a straightforward way to classify papers that study a wide range of attacks against machine learning, but it is also a guideline of what has been proposed so far in the literature. Yet, it offers a new perspective of breaking down the various types of features leveraged to classify different attacks by introducing a number of phases that each of the attack features are uniquely associated with.

The survey by Zhou et al. [22] has focused specifically on the relatively small number of approaches that model adversarial machine learning behaviour in a game-theoretic manner, where the players are the learning system and an adversary who is attacking it. Other surveys have emphasized on the particular characteristics of the application that is affected. For example, Akhtar et al. [23] have addressed adversarial attacks against deep learning in computer vision, investigating attacks that are specific to autoencoders and generative models, recurrent neural networks, deep reinforcement learning, semantic segmentation and object detection, and face attributes. In addition, Zhang et al.[24] focused on Deep Neural Networks (DNNs) and artificial intelligence, surveying methods of generating adversarial along with countermeasures. Duddu [25] assume that the adversary aims to reveal sensitive information handled by machine learning as well as information about the system architecture. They described a cyber-warfare testbed to test the effectiveness of various attack-defence strategies. Sun

et al. [26] tried to bridge the gap between theory and practice, formalized the threat model for practical attack schemes with adversarial examples against real systems. Most recently, Biggio et al. [27] have produced a very useful technical review of the evolution of active research in adversarial machine learning over the past ten years, while Yuan et al. [28] focused on the adversarial attacks and defences in models built with deep neural networks, proposing a taxonomy on attack approaches. The difference of our work is that we propose a taxonomy to classify the different papers that propose attacks against machine learning in order to unify the field. Our work can serve as a stepping stone towards the creation of a holistic defending framework and also motivate the creation of a taxonomy of defences against attacks on machine learning.

1.3. Attack models

The attack models we consider in this paper implement the following types of attacks: *poisoning* and *evasion*. The high level goal of these models is to maximize the generalization error of the classification and possibly mislead the decision making system towards desired malicious measurement values. As stated in [29], a system that uses machine learning aims to find a hypothesis function f that maps observable events, into different classes.

Let us consider a system that monitors network behaviour and performs anomaly-based intrusion detection. An instance of this behaviour is an event that is classified using utility function f either as Normal or Malicious. Let us assume an input space $\mathcal{X} = \{x_i\}$ and an output space $\mathcal{Y} = \{y_i\}$, where x_i is an event and y_i is the output of this event determined by f, i.e. $f(x_i) = y_i$. We assume that the system has been trained using N samples that form the training set $\mathcal{S}$ and it has derived the *system perception*, denoted by $\hat{y}$. After the end of the training phase, the system receives new events from the actual environment and classifies them. We define this as the *run-time phase* of the system. For every new event $\hat{x}_i$, f gives a new output $f(\hat{x}_i) = \hat{y}_i$. We have the following cases:

- If $\hat{x}_i$ is malicious and the system does not recognize it as such (false negative)

there is a loss l caused to the system.

- If $\hat{x}_i$ is malicious and the system recognizes it as such (true positive) or it is not malicious then there is no loss to the system.
- If $\hat{x}_i$ is not malicious and the system recognizes it as such (false positive) then there is a loss λ.

The aim of the attacker is to maximize the impact the attack has to the system by maximizing $|f(\hat{x}_i) - y_i|$. Thus, a challenge of the system that defends is to find a utility function that minimizes the losses, measured as the distance of $f(\hat{x}_i)$ to the real output y_i. This function can be linear or nonlinear and be more complex in formulation as in [30].

Evasion attacks: The adversary can undertake an *evasion* attack against classification during the testing phase thus producing a wrong system perception. In this case, the goal of the adversary is to achieve misclassification of some data towards, for example, remaining stealthy or mimicking some desirable behaviour. With regards to *network anomaly-based detection*, an intrusion detection system (IDS) can be evaded by encoding the attack payload in such a way that the destination of the data is able to decode it but the IDS is not leading to a possible misclassification. Thus, the attacker can compromise the targeted system being spotted out by the IDS. An additional goal of the attacker could be to cause *concept drift* to the system leading to continuous system retraining, thus, significantly degrading its performance [31].

Poisoning attacks: The adversary can poison the training dataset. To achieve this, the adversary *derives* and *injects* a point to decrease the classification accuracy [32]. This attack has the ability to completely distort the classification function during its training thus allowing the attacker to define the classification of the system in any way she wishes. The magnitude of the classification error depends on the data the attacker has chosen to poison the training. With regards to the aforesaid example, the adversary may be able to create dataset of anomalous network-layer protocol behaviour and train an anomaly-based intrusion detection system with a labelled attack dataset as the groundtruth. As a result, the detector will not be able to recognize cyber attacks

against this network-layer protocol threatening the security of the underlying system. This attack could be tailored to also have a significant impact to the quality of a signature-based intrusion detection system, which is responsible, for example, for detecting malware infecting a system or an infrastructure.

For example, a particularly insidious attack in this category is the *backdoor or Trojan attack*, where the adversary carefully poisons the model by inserting a backdoor key to ensure it will perform well on standard training data and validation samples, but misbehaves only when a backdoor key is present [33]. Thus an attacker can selectively make a model misbehave by introducing backdoor keys once the model is deployed. For instance, consider the case of assistive driving in autonomous vehicles: a backdoor could cause the model to misclassify a stop sign as speed limit whenever a specific mark has been placed on the stop sign. However, the model would perform as expected on stop signs without this mark, making the backdoor difficult to detect since users do not know the backdoor key a priori.

~~We denote by b the benefit of the attacker from attacking the system, where $b = |y_i - \hat{y}_i|$, and by c the cost of the attacker, which may be associated with the amount of effort the attacker has to put to perform the attack and the risk to get captured. If we assume that the attacker injects a malicious point x_c into a legitimate dataset D to form a new compromised dataset D'. The goal of the attacker is to maximize the generalization error W during the validation of m data samples, e.g.,: $\max_{x_c} W = \sum_{j=1}^{m} l(y_j, f(\hat{x}_j, W_j)) + \lambda||W||$ where f is the hypothesis function (i.e., classifier) that has been trained on a contaminated training set, which includes x_c, λ is a regularization co-efficient and $\hat{y}_j = W\hat{x}_j$.~~

The rest of the paper is organized as follows: Section 2 briefly describes the taxonomy of attacks against machine learning; Section 3 provides a systematic review of the adversarial machine learning approaches regarding each application area; Section 4 discusses the distribution of papers per taxonomy phase, open problems in adversarial machine learning and draws the conclusions giving some pointers for future work.

2. Proposed taxonomy

While the implementation details of attacks against machine learning may vary considerably, their individual steps can be broadly classified into two distinct phases: (i) Preparation and (ii) Manifestation, as illustrated in Figure 2 and detailed below. In the rest of this section, we discuss the different features

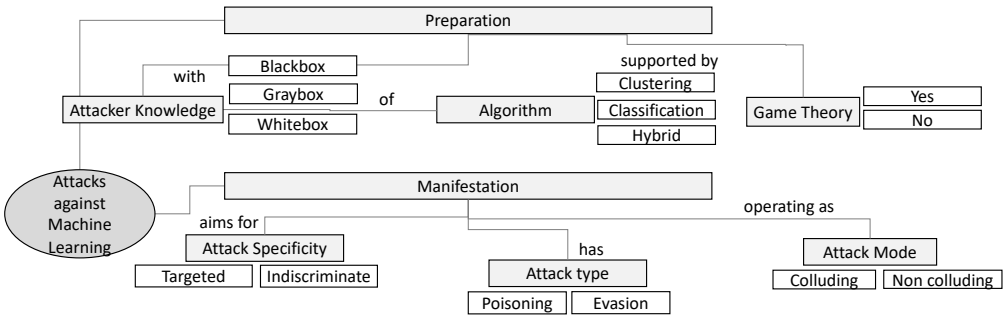

Figure 2: A taxonomy of adversarial attacks on machine learning.

of each taxonomy phase.

2.1. Preparation

In this phase, the attackers identify their resources and gather the intelligence required to prepare an attack plan. Here, what determines the characteristics of an adversarial machine learning approach is the knowledge required by the attacker, as well as the type of machine learning technique targeted and whether the attacker is strategic, i.e. they use game-theoretic techniques. As such we discuss the following features:

- **Attacker Knowledge**: Here, we take the simplified view whereby the attacker may know (K1) the *Ground truth*, (K2) the learning algorithm, or both, leading to the following attacker knowledge categories:
  - Blackbox attacks: $\neg K_1 \wedge \neg K2$.
  - Graybox attacks: $K1 \vee K2$.
  - Whitebox attacks: $K1 \wedge K2$.

According to [27], the attacker knowledge may refer to (i) the training data, (ii) the feature set, (iii) the machine learning algorithm along with the objective function minimized during training and (iv) any trained parameters if applicable.

- **Algorithm**: A large variety of machine learning techniques has been targeted in the literature. Indicatively, it is DNNs and Convolutional Neural Networks (CNNs) that are commonly addressed in the image recognition domain, while in spam email detection, more common are Naive Bayes, Support Vector Machines (SVM) and Logistic Regression (LR). Other techniques, such as K-Means, K-Nearest Neighbour (KNN), Linear Regression, Community Discovery and Singular Value Decomposition, are typically seen in the malware detection, biometric recognition and network failure and security breach detection domains. For the purposes of this taxonomy, we have classified the techniques based on the machine learning algorithm used in: i) clustering, ii) classification, or iii) hybrid fashion.
- **Game theory**: Adversarial machine learning is commonly equipped with a strategic element, whereby, in game theory terminology, the defender is the machine learning classifier and the attacker is a data generator aiming to contaminate, for example, the training dataset. Both choose their actions strategically in what can be seen as a non-cooperative game [34]. The adversary aims at confusing the classification or clustering with costs related to, e.g. the transformation process or probability of being detected. On the other hand, the defender incurs, for instance, a cost for misclassifying samples. The importance of game theory for the defender lies within the field of making the classifiers more aware of adversarial actions and more resistant to them.

2.2. Manifestation

This is the phase where the adversary launches the attack against the machine learning system. Largely dependent on the intelligence gathered in the preparation phase, the attack manifestation can be characterized based on the following characteristics:

- **Attack Specificity**: This refers to range of data points that are targeted by the attacker [29, 35]. It is also mentioned as *error specificity* in the recent survey of Barreno et al. [27].
  - **Targeted**: The focus of the attack is on a particular sample (e.g., specific spam email misclassified as legitimate) or a small set of samples.
  - **Indiscriminate**: The adversary attacks a very general class of samples, such as "any false negative" (e.g., maximizing the percentage of spam emails misclassified as legitimate).
- **Attack Type**: This refers to how the machine learning system is affected by an attack [29, 35].
  - **Poisoning**: Poisoning attacks alter the training process through influence over the *training* data.
  - **Evasion**: Evasion attacks exploit misclassifications but *do not affect training* (e.g. the learner or offline analysis, to discover information).
- **Attack Mode**: The original assumption of adversarial machine learning, which is still taken in most related literature, is that attackers work on their own (*non-colluding* case). The alternative is that different *colluding* attackers can collaborate, not only to cover their tracks but also to increase efficiency.

2.3. Attack Evaluation

The output of an attack's manifestation is primarily characterized by the nature of its impact on the *accuracy* of a machine learning approach. Each paper evaluates this impact by taking different approaches and metrics used to quantify and express it.

- **Evaluation Approach**: A goal of this work is to help researchers and developers improve the resilience of their mechanisms against adversarial machine learning by adopting approaches that have been thoroughly evaluated. We classify the related literature based on whether the proposed approaches have been evaluated *analytically*, in *simulation*, or *experimentally*.
- **Performance impact**: The primary aim of adversarial machine learning is to reduce the performance of a classification or clustering process that is

based on machine learning. For classification problems, this can be interpreted as increase in false positives, in false negatives, or in both. For clustering problems, the aim is generally to reduce accuracy.

- **False positives**: In classification problems, such as spam detection, where there are two states (spam or normal), the aim of an attacker may be to make the targeted system falsely label many normal emails as spam emails. This would lead to the user missing emails.
- **False negatives**: Using the same example, if the attacker aims to increase the false negatives, then many spam emails would go through the user's filters.
- **Both false positives and false negatives**: Here, the attacker aims to reduce the overall confidence of the user in their spam filtering system by letting spam emails go through and by filtering out normal emails.
- **Clustering accuracy reduction**: Compared to classification, the accuracy of clustering is less straight-forward to evaluate. Here, we include a general reduction of accuracy as the overall aim of the attacker of a clustering algorithm.

3. Adversarial attacks on machine learning

In this section we elaborate on the different adversarial attacks against applications that deploy machine learning. We have grouped the various articles based on the application domain so that it becomes clear how adversarial machine learning has evolved in each of these fields.

Fig. 3 illustrates the percentage of papers investigating adversarial machine learning for the various application domains. With the name "Other" we refer to articles that do not fall in any of the popular, within the field of adversarial machine learning, application domains (i.e., Intrusion Detection, Spam Filtering, Visual Recognition) while "Multipurpose" refers to papers that have investigated more than one application domain in order to assess their contributions.

Statistics show that the "Multipurpose" category covers the highest percent-

age of papers demonstrating the tendency of most authors to evaluate their work using different applications domains and understand the effect of the domain to the performance evaluation results. We observe that visual recognition is the second most investigated domain, which largely is due to the existence and the ease of use of well-known datasets such as the MNIST database of handwritten digits [36], ImageNet database [37].

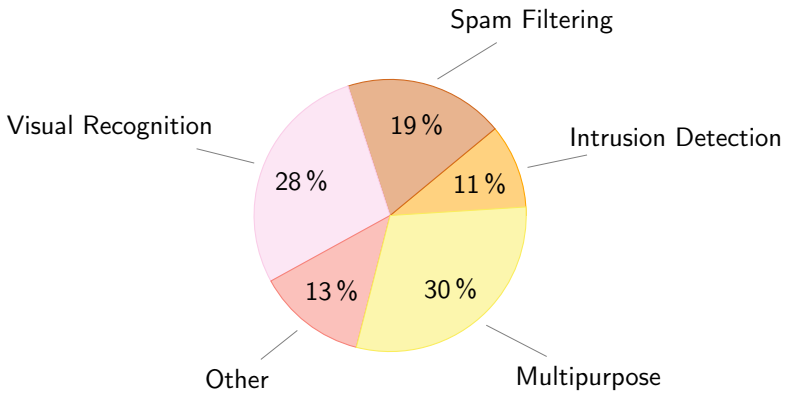

Figure 3: Percent of papers per application domain.

In the rest of this section, we discuss and present the surveyed papers per application domain for which we include three tables to provide a summary of the main features of the papers reviewed per attack phase (Preparation and Manifestation).

3.1. Intrusion Detection

3.1.1. Naive Learning Algorithm

Barreno et al. were the first who gave relevant properties, such as attack influence and specificity, for analyzing attacks on machine learning systems. In [29], they provided an example from the intrusion detection systems domain, in which an adversary can launch a whitebox *poisoning* attack by maliciously miss-training a learning IDS system, thus making the classifier (*classification*) unable to determine if a specific sample (*targeted*) is malicious or not (incurring *both false positives and false negatives*).

3.1.2. Support Vector Machine (SVM)

Biggio et al. [38] created a simple algorithm for evasion of classifiers with differentiable discriminant functions. They investigated the attack effectiveness on a realistic application related to the detection of PDF malware, and empirically showed that very popular classification algorithms (in particular, SVMs and neural networks) can still be evaded (*evasion* attack) by launching graybox to whitebox attacks against the classifier with high probability even if the adversary can only learn a copy of the classifier (*classification*) from a small surrogate data set. The attacker's goal would be to manipulate a single sample to be misclassified leading to increasing rate of *false negatives*.

3.1.3. Least Absolute Shrinkage and Selection Operator - Elastic Net

Xiao et al. [39] also performed experiments on PDF malware detection and were one of the first groups that proposed a framework to categorize different attacks of embedded feature selection algorithms using previously proposed poisoning and evasion attack models for the security of classification and clustering algorithms (*classification* and *clustering*). The attacker's goal, who launches graybox to whitebox attacks in these settings is to poison (*poisoning* attack) the training data, in both *targeted* and *indiscriminate* ways, so that a wrong subset of features is selected. The authors derived a poisoning feature selection algorithm increasing either *both false positives and false negatives* or causing *clustering accuracy reduction*.

3.1.4. Spectral Clustering - Community Discovery AND node2vec

Chen et al. [40] focus on another problem that may arise in graph based detection systems used in network detection. They devised two novel graybox and whitebox attacks launched by adversaries who use targeted noise injection (*poisoning*) and small community attacks against graph clustering or embedding techniques with their primary purpose to avoid detection (increase *false*

	Att. Knowl.			Algorithm			Adv. Strat.	
Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.
Barreno et al. [29]	✗	✗	✓	✗	✓	✗	✗	✓
Biggio et al. [38]	✗	✓	✓	✗	✓	✗	✗	✓
Xiao et al. [39]	✗	✓	✓	✓	✓	✗	✗	✓
Chen et al. [40]	✗	✓	✗	✓	✗	✗	✗	✓
Rubinstein et al. [41]	✓	✓	✓	✓	✗	✗	✗	✓
Wang et al. [42]	✗	✓	✓	✓	✗	✗	✗	✓
Demetrio et al. [43]	✓	✗	✗	✗	✓	✗	✗	✓

Table 1: Comparative analysis of Intrusion Detection Systems at Preparation Phase

negatives). The latter include community discovery, Singular Value Decomposition and node2vec, which are a combination of *classification* and *clustering*. The noise injection is *targeted* while the small community attacks are *indiscriminate* as a result of the attacker choosing, in s random manner, graph nodes to manipulate.

3.1.5. Principal Component Analysis

Rubinstein et al. [41] focused on detecting anomalies in backbone networks through *clustering* and considered a new model of Boiling frog attack schemes using the *Principal Component Analysis* (PCA-subspace method). In these *poisoning* attack schemes, the attacker performs *blackbox* to whitebox attacks against the learner, poisoning the training data slowly, but increasingly, over weeks remaining undetected. The attack focuses on the training phase of the system and increases *both false positives and false negatives* rate with its final goal being to cause Denial of Service. This work assumed two cases: (i) the attacker performs link poisoning when the traffic volume on the link exceeds a parameter (*targeted*) in order to increase traffic variance and (ii) the attacker

	Attack Specif.		Attack Mode		Attack Type	
Ref.	Targeted	Indiscrim.	Collud.	Non-collud.	Poisoning	Evasion
Barreno et al. [29]	✓	✗	✗	✓	✓	✗
Biggio et al. [38]	✓	✓	✗	✓	✗	✓
Xiao et al. [39]	✓	✓	✗	✓	✓	✗
Chen et al. [40]	✓	✓	✗	✓	✓	✗
Rubinstein et al. [41]	✓	✓	✗	✓	✓	✗
Wang et al. [42]	✓	✓	✓	✓	✓	✗
Demetrio et al. [43]	✓	✗	✗	✓	✗	✓

Table 2: Comparative analysis of Intrusion Detection Systems at Manifestation Phase

injects traffic on any link (*indiscriminate*).

3.1.6. Support Vector Machine - Bayesian - Decision Trees and Random Forests

Wang et al. [42] performed an empirical study of the efficacy of ML models (*clustering*) for detecting malicious crowd-sourcing, also known as crowd-turfing. They evaluated a powerful class of whitebox *poisoning* attacks, where the adversary injects: (i) either carefully selected data (*targeted*) into training data to greatly reduce the efficacy of the detector, increasing the rate of *both false positives and false negatives*; or (ii) random normal accounts to the turfing class (indiscriminate). Furthermore, the evasion attacks studied by the authors are *targeted* since only individual instances (represented by the "workers") can be altered.

3.1.7. Convolutional Neural Networks (CNN)

Within the context of deep neural networks as applied to binary malware detection, Demetrio et al. [43] proposed a *blackbox evasion* attack against the *classification* capabilities of a convolutional neural network named MalConv [44]. They applied the integrated gradients technique to malware programs.

	Eval. Approach			Perform. Impact			
Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.
Barreno et al. [29]	✗	✗	✗	✗	✗	✓	✗
Biggio et al. [38]	✗	✗	✓	✗	✓	✗	✗
Xiao et al. [39]	✗	✗	✓	✗	✗	✗	✓
Chen et al. [40]	✗	✗	✓	✗	✓	✗	✗
Rubinstein et al. [41]	✗	✗	✓	✗	✗	✓	✗
Wang et al. [42]	✗	✗	✓	✗	✗	✓	✗
Demetrio et al. [43]	✗	✗	✓	✗	✓	✗	✗

Table 3: Comparative analysis of Intrusion Detection Systems' Attack Evaluation

Their methodology selects the closest byte to the embedded space of binary aiming at increasing the probability of evasion thus increasing *false negatives*).

3.2. Spam Filtering

3.2.1. Linear Classifiers

Focusing on spam detection and the use of Naive Bayes, Lowd and Meek [45] defined an Adversarial Classifier Reverse Engineering (ACRE) model. In ACRE, the adversary launches graybox *evasion* attacks against the classifier (*classification*) by sending membership queries to the classifier. By attacking the system's testing phase, the goal of the attacker is to determine whether a specific instance is malicious or not. Additionally, the attack aims at increasing the rate of *false negatives* by classifying spam messages as benign. The authors assume that the attacker optimizes his cost over all instances characterizing the attack as *indiscriminate*.

3.2.2. Game Theory

Li and Vorobeychik [46] proposed a similar approach in terms of attack influence (*evasion* attack) and used feature reduction in a specific phase to increase the rate of *false negatives*. They argued that the graybox attacks launched by an adversary depends on the restricted budget he has, and so they modeled a Stackelberg Game with multiple attackers, where the Learner moves first by choosing a linear classifier, and all attackers simultaneously and independently respond to the learner's choice by manipulating the feature vector, where the training dataset is the pair of (feature vectors, corresponding binary labels). This is an *indiscriminate* attack for which the attacker solves an optimization problem over the entire set of instances.

Bruckner and Scheffer [47] in the context of spam email filtering study Stackelberg games in which the learner acts as the leader committing to a predictive model, while the adversary (data generator) acts as the follower generating the test data at application time. According to their threat model, in a *classification* the adversary launches graybox attacks by influencing the generation of the data at application time trying to impose the highest cost on the learner (*evasion*) with final goal to make spam email misclassified as benign (*increase false negatives*). This is an *indiscriminate* attack as the attacker transforms the entire training distribution. The same author and his team [48] worked on email spam filtering through the use of *Support Vector Machines* (*classification*) and created a model of adversarial learning problems as static games. They proposed that the adversary performs graybox data *poisoning* during training and test time. The described attack can increase *both false positives and false negatives* depending on if only spam emails are misclassified or every email is misclassified regardless its content.

Liu and Chawla [49] created a model of the interaction between an email spammer and the defender as a two-player infinite Stackelberg game, where players have two actions: Spam or Status Quo (attacker) and Retrain or Status Quo (defender). When *Spam* is chosen, the adversary attacks the classifier by

actively modifying spam emails in order to get through the filter. When *Retrain* is chosen, the classifier chooses to retrain its system to lower the error rate. *Status quo* is a strategy according to which, the attacker does nothing to bypass the spam filters while the defender does retrain its system tolerating a potential increase in spam emails getting through. The attacker is launching whitebox *poisoning* attacks targeting the training process of the classifier with the aim to increase the *false negative* rate. They proved that the players' payoff functions, regarding a two class classification (*classification*), are given by the solution of linear optimization problems (*indiscriminate*) which are solved by backward induction. Zhang and Zhu [50] established a game-theoretic framework to capture the conflicting interests between the adversary and a set of distributed data processing units. According to their threat model an whitebox attack can impact the training process (*targeted*) by modifying the training data (*poisoning* attack) of the SVM (*classification* having as purpose to maximize the error rate (increase *both false positives and false negatives*) by choosing strategies that maximize the attacker's payoff over the entire set of data instances (*indiscriminate*).

As far as regression (*classification*) problems in spam detection are concerned, Grosshans et al. [51] study non zero sum games with incomplete information about the adversary's cost function. In the Bayesian game they relax the full information game into an asymmetric one where the adversary has full information (whitebox attack) about the learner's cost function, while the learner's information is expressed by a distribution over the parameters of the adversary's cost function, due to uncertainty. In their threat model the adversary can exercise some control over the distribution of the data (*evasion* attack) at application time having as goal to misclassify his data as benign (increase *false negatives*). As the attacker generates an entire training matrix, this is an *indiscriminate* attack.

3.2.3. Naive Bayes

Dalvi et al. [30] also conducted experiments in the *spam detection* domain, focusing on a Naive Bayes classifier (*classification*). Their approach differs in that the adversary deploys an optimal feature-changing strategy against the classifier in the context of a two-player game. The strategy is to cause the classifier to classify a number of positive instances as negative (*false negatives*) by modifying them during training. The type is perceived as whitebox *evasion* as the adversary constantly updates the approach according to the countermeasures of the opponent who improves the classifier. Given that the attack strategy is over all instances this is characterized as a *indiscriminate* attack.

Following the same approach regarding Naive Bayes applied on spam detection, Barreno et al. [35] investigated four hypothetical attack scenarios. From these, the Spam Foretold and the Rogue Filter attacks are *poisoning* and the Shifty Spammer and the Mistaken Identity attacks are *evasion*. They discussed both *targeted* and *indiscriminate* versions of these attacks. In all cases, the adversary launches graybox and whitebox attacks against the classification mechanism (*classification*) in order to increase the rate of *both false positives and false negatives*. This is formulated in an adversarial machine learning game, in which the defender chooses a learning algorithm and the attacker chooses a procedure for selecting distributions from which infected training datasets are generated.

Nelson et al. [52] also studied *poisoning* attacks through the use of *Naive Bayes* (*classification*). In their dictionary attack, the adversary sends attack emails that contain many words likely to occur in legitimate email correspondence (*indiscriminate*). In their focused attack, the adversary sends attack emails to the victim containing specific words likely to occur in the target email (*targeted*). The same attack may have different levels of knowledge about the target email ranging from blackbox to whitebox and aims at rendering the spam filter unusable (increasing the rate of both *false positives and false negatives*).

Naveiro et al. [53] followed a different approach by proposing the use of risk analysis in adversarial classification (*classification*). Specifically, their alternative methodology, is called Adversarial Classification Risk Analysis (ACRA) where no assumptions of common knowledge are made. The adversaries trie

	Att. Knowl.			Algorithm			Adv. Strat.			Att. Knowl.			Algorithm			Adv. Strat.	
Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.	Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.
Lowd et al.[45]	✗	✓	✗	✗	✓	✗	✗	✓	Li et al. [46]	✗	✓	✓	✗	✓	✗	✓	✗
Dalvi et al. [30]	✗	✗	✓	✗	✓	✗	✓	✗	Barreno et al. [35]	✗	✓	✗	✗	✓	✗	✓	✗
Nelson et al. [52]	✓	✓	✓	✗	✓	✗	✗	✓	Bruckner et al. [47]	✗	✓	✗	✗	✓	✗	✓	✗
Bruckner et al. [48]	✗	✓	✗	✗	✓	✗	✓	✗	Liu et al. [49]	✗	✗	✓	✗	✓	✗	✓	✗
Zhang et al. [50]	✗	✗	✓	✗	✓	✗	✓	✗	Grosshans et al. [51]	✗	✗	✓	✗	✓	✗	✓	✗
Xiao et al. [54]	✗	✗	✓	✗	✓	✗	✗	✓	Naveiro et al. [53]	✗	✓	✗	✗	✓	✗	✗	✓

Table 4: Comparative analysis of Spam Filtering at Preparation Phase

to make their own spam emails undetectable (increase of *false negatives*) from the classifier by adapting to the classifier's responses (graybox *evasion* attack) through decision making. By having an attacker who maximizes their expected utility, the attack is characterized as *indiscriminate*.

	Eval. Approach			Perform. Impact					Eval. Approach			Perform. Impact				
Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.	Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.
Lowd et al.[45]	✓	✗	✓	✗	✓	✗	✗	Li et al. [46]	✓	✗	✓	✗	✓	✗	✗
Dalvi et al. [30]	✓	✗	✓	✗	✓	✗	✗	Barreno et al. [35]	✗	✗	✓	✗	✗	✓	✗
Nelson et al. [52]	✗	✗	✓	✗	✗	✓	✗	Bruckner et al. [47]	✓	✗	✓	✗	✓	✗	✗
Bruckner et al. [48]	✓	✗	✗	✗	✗	✓	✗	Liu et al. [49]	✓	✗	✓	✗	✓	✗	✗
Zhang et al. [50]	✓	✗	✓	✗	✗	✓	✗	Grosshans et al. [51]	✓	✗	✓	✗	✓	✗	✗
Xiao et al. [54]	✓	✗	✓	✗	✗	✓	✗	Naveiro et al. [53]	✓	✗	✓	✗	✓	✗	✗

Table 6: Comparative analysis of Spam Filtering's Attack Evaluation

	Att. Spec.		Att. Mode		Att. Type			Att. Spec.		Att. Mode		Att. Type	
Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion	Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion
Lowd et al.[45]	✗	✓	✗	✓	✗	✓	Li et al. [46]	✗	✓	✓	✓	✗	✓
Dalvi et al. [30]	✗	✓	✗	✓	✗	✓	Barreno et al. [35]	✓	✓	✗	✓	✓	✓
Nelson et al. [52]	✓	✓	✗	✓	✓	✗	Bruckner et al. [47]	✗	✓	✗	✓	✗	✓
Bruckner et al. [48]	✗	✓	✗	✓	✓	✗	Liu et al. [49]	✗	✓	✗	✓	✓	✗
Zhang et al. [50]	✗	✓	✗	✓	✓	✗	Grosshans et al. [51]	✗	✓	✗	✓	✗	✓
Xiao et al. [54]	✓	✗	✗	✓	✓	✗	Naveiro et al. [53]	✗	✓	✗	✓	✗	✓

Table 5: Comparative analysis of Spam Filtering at Manifestation Phase

3.2.4. Support Vector Machine (SVM)

Xiao et al. [54] studied a poisoning attack in an SVM (*classification*) spam email detection model (*targeted*). In this paper, the attacker is able to flip a bounded number of samples (*targeted*) in the training data (whitebox *poisoning* attack), whose values are picked from an also bounded distribution. The goal of the attacker is the maximization of the classification error (*both false positives and false negatives*). Due to the computational hardness of solving the bounded problem optimally, the authors proposed two heuristics to derive efficient approximate solutions (attack strategies). The first heuristic is based on the generation of different sets of flips and the selection of the one that achieves the best value. The second heuristic is a greedy BFS algorithm that generates correlated subsets of label flips and the selection of that set that after a number of iterations maximizes the empirical risk.

3.3. Visual Recognition

3.3.1. Principal Component Analysis (PCA)

Carlini and Wagner [55] have also investigated neural networks applied to image *classification*. They have studied three types of threat models based on the extent of knowledge of the adversary: a perfect-Knowledge Adversary; limited-Knowledge Adversary; and a zero-Knowledge Adversary. Each leads to a different attack: whitebox attack, graybox attack, and a blackbox attack that the authors have implemented and can be either *evasion* or *poisoning* attack. The adversary attacks different phases (training or testing), depending on his knowledge, with the goal to increase *both false positives and false negatives*. The authors focus on targeted adversarial examples exclusively in this paper. Finally, they have used two datasets; the MNIST dataset [56] and the CIFAR-10 dataset [57].

3.3.2. Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbor

Hayes and Danezis [58] also worked in Neural Networks and focused on launching blackbox attacks against a variety of machine learning models like Random Forest, SVM, and K-Nearest Neighbor (*classification*) hosted in an amazon instance by creating adversarial examples and testing their effects on the classifier (*evasion* attack). The final goal of the attacks is to increase *both false positives and false negatives*. In order to create their adversarial examples they make use of the MNIST dataset and use them to augment the robustness of a classifier against adversarial examples, something that they achieve.

3.3.3. Artificial Neural Network (ANN)

Shaham et al. [15] proposed a training framework to increase the stability and the robustness of artificial neural networks against adversarial examples and performed experiments in visual recognition to test their assumption. According to them, an adversary launches whitebox attacks against the classifier (*classification*), by targeting a specific artificial neural network and aims to construct a perturbation of the original input data (*poisoning* attack). In this data the network parameters are updated in way that leads to the misclassification of the

training examples (increase *both false positives and false negatives*). Adversarial examples are generated for a given training point to solve the box-constrained optimization problem (*indiscriminate*).

	Att. Knowl.			Learn.			Adv. Strat.			Att. Knowl.			Learn.			Adv. Strat.	
Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.	Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.
Biggio et al. [59]	✗	✗	✓	✓	✗	✗	✗	✓	Szegedy et al.[17]	✗	✓	✗	✗	✓	✗	✗	✓
Goodfellow et al.[60]	✗	✗	✓	✗	✓	✗	✗	✓	Goodfellow et al. [61]	✗	✗	✓	✗	✓	✗	✓	✗
Carlini et al. [55]	✓	✓	✓	✗	✓	✗	✗	✓	Hayes et al. [58]	✓	✗	✗	✗	✓	✗	✗	✓
Hayes et al. [62]	✗	✗	✓	✗	✓	✗	✗	✓	Nguyen et al. [16]	✓	✓	✓	✗	✓	✗	✗	✓
Papernot et al. [63]	✓	✗	✗	✗	✓	✗	✗	✓	Papernot et al. [64]	✓	✗	✗	✗	✓	✗	✗	✓
Kurakin et al. [65]	✓	✓	✓	✓	✓	✗	✗	✓	Shaham et al. [15]	✗	✗	✓	✗	✓	✗	✗	✓
Evtimov et al. [66]	✗	✗	✓	✗	✓	✗	✗	✓	Radford et al. [67]	✗	✓	✓	✓	✓	✗	✗	✓
Schottle et al. [68]	✗	✓	✗	✗	✓	✗	✗	✓	Chivikula et al. [69]	✓	✗	✗	✗	✓	✗	✓	✗
Madry et al. [70]	✓	✓	✓	✗	✓	✗	✗	✓	Springenberg et al. [71]	✗	✓	✗	✗	✓	✓	✗	✓
Athalye et al. [72]	✗	✗	✓	✗	✓	✗	✗	✓	Dong et al. [73]	✓	✗	✗	✗	✓	✗	✗	✓
Jeong et al. [74]	✓	✗	✗	✗	✓	✗	✗	✓									

Table 7: Comparative analysis of Visual Recognition at Preparation Phase

3.3.4. Convolutional neural networks (CNN)

Radford et al. [67] introduced a new class of architectures of convolutional networks (CNNs) that they call deep convolutional generative adversarial networks (DCGANs). They used trained discriminators for image classification tasks. They showed that DCGANs have competitive performance with unsupervised algorithms, proving that a methodology which was commonly used in *classification* can be also effective even in an *clustering* setting. In that paper, the adversaries vary in terms of knowledge (graybox-whitebox attacks) and they can launch a wide variety of attacks in terms of attack influence (*evasion* and *poisoning* attacks) and machine learning phase being attacked with the aim to ei-

ther increase *both false positives and false negatives* or cause *clustering accuracy reduction*. Attacks take place across multiple samples leading to *indiscriminate* nature.

Schottle et al. [68] used methods from multimedia forensics to bring robustness to CNNs against adversarial examples. In this work, the attacker launches graybox attacks against the classifier (*classification*) by using Projected Gradient Descent to create adversarial examples and test (*evasion* attack) how they are classified to achieve maximization of the misclassification of benign samples as malicious (increase *false positives*) leading to a *indiscriminate* attack. In order to defend against that group of attackers, the authors combine forensic and steganalytic methodologies to increase the robustness of the classifier experimenting with the MNIST database, eventually getting good results.

Chivukula et al. [69], using the MNIST handwritten images database propose a genetic algorithm for deep learning on Convolutional Neural Networks (*classification*) that is formulated as a two player zero sum sequential Stackelberg game. In every iteration of the game, the adversary, who has no prior knowledge about the networks structure (blackbox attacks), targets the learner with data produced using genetic operators with the goal to alter many positive labels to negative (increase *false negatives*). The learner adapts to these adversarial data by retraining (*poisoning* attack) the weights of the CNN layers. Upon convergence of the game, an adversarial manipulation parameter is returned by the algorithm which is then imported into the original data to produce an adversarial sample for the retraining (*targeted*).

3.3.5. Generative Adversarial Networks (GAN)

Springenberg [71] studied the problem of an adversarial generative model that regularizes a discriminatively trained classifier (*clustering* and *hybrid*). He proposed an algorithm which is the result of a natural generalization of the generative adversarial networks (GAN) framework as an extension of the regularized information maximization (RIM) framework (*indiscriminate*) to robust

	Att. Spec.		Att. Mode		Att. Type			Att. Spec.		Att. Mode		Att. Type	
Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion	Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	*Poisoning*	*Evasion*
Biggio et al. [59]	✓	✓	✗	✓	✓	✓	Szegedy et al.[17]	✗	✓	✗	✓	✓	✗
Goodfellow et al.[60]	✗	✓	✗	✓	✓	✗	Goodfellow et al. [61]	✗	✓	✗	✓	✗	✓
Carlini et al. [55]	✓	✗	✗	✓	✓	✓	Hayes et al. [58]	✓	✓	✗	✓	✗	✓
Hayes et al. [62]	✓	✓	✗	✓	✓	✗	Nguyen et al. [16]	✗	✓	✗	✓	✗	✓
Papernot et al. [63]	✗	✓	✓	✗	✗	✓	Papernot et al. [64]	✗	✓	✓	✗	✗	✓
Kurakin et al. [65]	✗	✓	✗	✓	✓	✓	Shaham et al. [15]	✗	✓	✗	✓	✓	✗
Evtimov et al. [66]	✓	✓	✗	✓	✗	✓	Radford et al. [67]	✗	✓	✓	✗	✓	✓
Schottle et al. [68]	✗	✓	✗	✓	✗	✓	Chivikula et al. [69]	✓	✗	✗	✓	✓	✗
Madry et al. [70]	✗	✓	✗	✓	✓	✓	Springenberg et al. [71]	✗	✓	✗	✓	✓	✗
Athalye et al. [72]	✓	✓	✗	✓	✓	✗	Dong et al. [73]	✓	✓	✗	✓	✗	✓
Jeong et al. [74]	✗	✓	✗	✓	✓	✗							

Table 8: Comparative analysis of Visual Recognition at Manifestation Phase

classification against an optimal adversary. According to his point of view, an adversary is able to launch *targeted* graybox *poisoning* attacks against the classifier in order to avoid detection (increase *false negatives* rate). Generative adversarial networks are commonly described as a two-player game [61]. In [71], in each step the generator (attacker) produces an example from random noise that has the potential to confuse the discriminator (defender). The latter then receives some real data along with samples produced by the attacker, and attempts to classify them as legitimate or malicious.

	Eval. Appr.			Perform. Impact					Eval. Appr.			Perform. Impact			
Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.	Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.
Biggio et al. [59]	✗	✗	✓	✗	✗	✗	✓	Szegedy et al.[17]	✗	✗	✓	✗	✗	✓	✗
Goodfellow et al.[60]	✗	✗	✓	✗	✗	✓	✗	Goodfellow et al. [61]	✓	✗	✓	✗	✓	✗	✗
Carlini et al. [55]	✗	✗	✓	✗	✗	✓	✗	Hayes et al. [58]	✗	✗	✓	✗	✗	✓	✗
Hayes et al. [62]	✗	✗	✓	✗	✓	✗	✗	Nguyen et al. [16]	✗	✗	✓	✗	✓	✗	✗
Papernot et al. [63]	✗	✗	✓	✗	✓	✗	✗	Papernot et al. [64]	✗	✗	✓	✗	✓	✗	✗
Kurakin et al. [65]	✗	✗	✓	✗	✗	✓	✗	Shaham et al. [15]	✓	✗	✓	✗	✗	✓	✗
Evtimov et al. [66]	✗	✗	✓	✗	✗	✓	✗	Radford et al. [67]	✓	✗	✓	✗	✗	✓	✓
Schottle et al. [68]	✗	✗	✓	✓	✗	✗	✗	Chivikula et al. [69]	✓	✗	✓	✗	✓	✗	✗
Madry et al. [70]	✗	✗	✓	✗	✗	✓	✗	Springenberg et al. [71]	✗	✗	✓	✗	✓	✗	✗
Athalye et al. [72]	✓	✗	✓	✗	✗	✓	✗	Dong et al. [73]	✓	✗	✓	✗	✓	✗	✗
Jeong et al. [74]	✗	✗	✓	✗	✗	✓	✗								

Table 9: Comparative analysis of Visual Recognition's Attack Evaluation

3.3.6. Deep Learning

Biggio et al. [59] studied obfuscation (*targeted*) and poisoning (*indiscriminate*) attacks against *clustering* ML algorithms focused on handwritten digit recognition where the adversaries are thought to launch whitebox attacks the clustering mechanism. The attack type is considered *poisoning* because the adversaries can manipulate part of the data to be clustered. In addition to that, the specific attacks cause *clustering accuracy reduction* by altering the clustering output on any data point. The same team also describes "practical" and "optimal" *evasion* attacks.

Szegedy et al. [17] focused on the loss of performance of DNNs when adversarial examples are included in the training sample (*poisoning* attack). According to their scenario, an adversary launches graybox attacks against the

classifier (*classification*) by performing small perturbations to images with the goal to make the system misclassify (increase *both false positives and false negatives*) the images. The perturbations can be found by optimizing the input to maximize the prediction error leading to an *indiscriminate* attack.

Goodfellow et al. [60] have shown that *neural networks* and *deep neural networks* are vulnerable to adversarial perturbation of the training set (*targeted*) because of their linear nature. Consequently, an adversary can launch a whitebox *poisoning* attack against the training set after adding perturbations to the input. This action will increase *both false positives and false negatives* of the classification mechanism (*classification*), which in their case was applied to image recognition. Finally, this is an *indiscriminate* attack as the adversary affects the entire set of instances. The same researchers [61] have proposed a new framework called "adversarial nets" for *deep learning*, which sidesteps the difficulties of approximating many intractable probabilistic computations that arise in maximum likelihood estimation and related strategies, and leverages the benefits of piece-wise linear units in the generative context. This framework corresponds to a minimax two-player game. The attacker creates a generative training model aiming to maximize the probability that the classifier will not detect his input (*false negative*). They have created an whitebox *evasion* attack that does not affect the training process and imitates the entire normal data distribution (*indiscriminate*) with an added noise.

Nguyen et al. [16] shed light to another scientific area of pattern recognition, the use of DNNs for visual classification (*classification*). They proved that an adversary launches blackbox to graybox attacks, by using evolutionary algorithms and gradient ascent is able to fool a DNN (increase *false negatives*). According to the authors retraining a DNN using fake images and labelling them as such makes no difference as the production of new sets of fake images (*evasion* attack) will confuse the classifier again. By using evolutionary algorithms, the adversary generates novel images launching an *indiscriminate* attack.

Papernot et al. [63] highlighted the importance of adversarial sample transferability in the context of image recognition using DNNs. According to their

attack scenario, the attackers launch blackbox attacks against the classifier (*classification*) and they use adversarial samples to mislead a model as well as models that are influenced by this. They define this as *adversarial sample transferability*. These attacks are launched towards a DNN and are often called black box attacks. Their type is *evasion* because the adversary's only capability is to observe labels assigned by the DNN and their goal is to increase the *false negative* rate of the DNN so as to misclassify adversarial inputs, while remaining correctly classified by a human observer. Since the attacker solves an optimization problem to produce a vector of adversarial instances, this is an *indiscriminate* attack. The same main author and his team [64] provided a method to craft adversarial samples using a substitute DNN. They assume that the attacker can launch blackbox attacks against the classifier (*classification*). The attack strategy includes building a substitute DNN approximating the oracle based on synthetic queries made directly to the oracle and then using it to craft adversarial samples that will later be used against the oracle (*evasion* attack). As the manifestation of the attack will cause the misclassification of the adversarial samples the increase of *false negatives* remains as the final goal. Since the attacker aims to produce a minimally altered version of any input this leads to an *indiscriminate* attack.

Kurakin et al. [65] injected adversarial examples into the training set using the adversarial training methodology in both *classification* and *clustering*. They also observed the transferability of adversarial examples between different models and the importance of the "label leaking" property. Their examples focused on ImageNet data set, varying in terms of attacker knowledge (blackbox - whitebox attacks), attack type (*poisoning* and *evasion* attack), as well as performance impact (*both false positives and false negatives* and *clustering accuracy reduction*. Their attack model is *indiscriminate* as it covers the entire set of pixels of an image.

Hayes and Danezis[62] investigated universal adversarial networks in the context of neural networks and image recognition. They studied generative networks that are capable of fooling a target classifier (*classification*) when its generated

output is added to a clean sample from a dataset. In their attack models, the attacker launches whitebox attacks against the classifier by accessing the training data (*poisoning* attack). In addition, the attacker wants an adversarial sample to be misclassified as something else, thus increasing *false negatives* rate. [58] and [62] investigated both targeted and indiscriminate attack cases but from the point of view of (i) having the attacker mapping any image to a chosen class (*targeted*); and (ii) performing any misclassification in the target model (*indiscriminate*).

Evtimov et al. [66] focused on real road sign recognition (*classification*) and the use of DNNs by introducing a new attack method called Robust Physical Perturbations. This generates perturbations by taking images under different conditions into account. The adversary can launch whitebox *evasion* attacks (checking the effects of his attacks) by using physical perturbations (*targeted*) with upper goal the misclassification (increase *both false positives and false negatives*). These perturbations are categorized in two types: subtle perturbations that occupy the entire region of the sign (can be seen as *indiscriminate*); and (2) camouflage perturbations that take the form of graffiti and abstract art (can be seen as *targeted*).

Madry et al. [70] studied the robustness of deep learning algorithms against such well-defined classes of adversaries. They claim that their work is an important step towards fully resistant deep learning models. More specifically, they studied the robustness of neural networks against adversaries through optimization in the context of computer vision. They provided guarantees on the level of security solving a maximin optimization problem and modelled both attacks against *classification* and defences through a general theoretical framework. Their theoretical and experimental results show that the locally maximum values of the loss function of an adversary found by a Projected Gradient Descent (PGD) are similar for normally and adversarially trained networks (*poisoning* and *evasion* attacks). This means that robust networks against PGD adversaries are also robust against a wide range of attacks, increasing *false positives*, *false negatives* or *both false positives and false negatives*. Their experiments used

MNIST and CIFAR10 datasets for whitebox and blackbox attacks. Due to the attacker solving the proposed inner maximazitation problem, the investigated attack is categorized as *indiscriminate*.

Recently, Athalye et al. [72] investigated the prevalence of obfuscated gradients on neural networks in MNIST [56] and CIFAR10 [75] datasets. They investigated *whitebox* attacks against the classifier (*classification*) generating both *targeted* and *indiscriminate* adversarial examples with iterative optimization-based methods that require gradients obtained through backpropagation. Consequently, many defenses either intentionally or unintentionally cause gradient descent to fail (*poisoning*) by either minimizing, or maximizing the classification loss increasing *both false positives and false negatives*. This happens due to obfuscated gradients caused by gradient shattering, stochastic gradients, or vanishing/exploding gradients.

In the context of deep neural networks, Dong et al. [73] proposed *targeted* and *indiscriminate blackbox* attacks. They used a broad class of momentum iterative gradient-based methods that generate adversarial examples. These can successfully fool robust *classification* models (*evasion*). In their experiments, they used the ImageNet database [37]. Their goal was to boost adversarial attacks, which can effectively fool whitebox as well as blackbox models increasing *false negatives*.

Jeong et al. [74] focused on the impact of adversarial samples on multimedia video surveillance using deep learning and *blackbox* settings. Using the MNIST and NSL-KDD datasets, they injected, in an *indiscriminate* way, adversarial samples into the training session of Autoencoder and Convolution Neural Network (CNN) *classification* models. This was a poisoning attack aiming to lower the accuracy of the classifier thus increasing *both false positives and false negatives*.

3.4. Other applications

3.4.1. Reinforcement Learning

Uther et al. [76] were one of the first teams that focused on multi-agent reinforcement learning techniques. In specific, they created a framework, consisting of a two-player hexagonal grid soccer in order to evaluate the algorithms proposed in this scientific area. According to their assumption the attacker launches graybox attacks against the classifier (*classification*) because he needs to collect large amounts of data by observing (*targeted*) his opponent's moves (*evasion* attack). His final goal is to increase the *false negatives* of the attacked classifier by solving a minimax mathematical problem (*indiscriminate*).

3.4.2. Collaborative Filtering

In the context of movie recommendation systems Guillory and Bilmes [77], have studied a submodular set cover problem where a set of movies is suggested to a user according to their feedback. The assumption is that the adversary attempts a graybox *poisoning* attack against the the *clustering* mechanism by injecting a limited amount of adversarial noise (*targeted*) during query learning. This attack aims at causing *clustering accuracy reduction*, resulting in bad recommendations if we take into consideration the user's preferences.

3.4.3. Recurrent Neural Networks (RNNs)

Papernot et al. [18] proposed an attack model for the misclassification of outputs of Recurrent Neural Networks (RNNs) by crafting adversarial sequences both categorical and sequential during test time (*evasion* attack). According to their perception the attacker launches *indiscriminate* graybox attacks (model's architecture, computational graph and values of parameters learned during training) during test phase. In their experiments based on a movie review classifier (*classification*), the authors derived 100% misclassification of the training data of categorical sequences (increase *both false positives and false negatives*), when they alter 13% of the words in movie reviews. As the attacker solves an optimization problem to determine the vector of adversarial instances, this is characterized as an *indiscriminate* attack.

	Attacker Knowl.			Learning			Advanced Strategy.	
Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.
Uther et al. [76]	✗	✓	✗	✗	✓	✗	✓	✗
Guillory et al. [77]	✗	✓	✗	✓	✗	✗	✗	✓
Papernot et al. [18]	✗	✓	✗	✗	✓	✗	✗	✓
Alfeld et al. [78]	✗	✓	✗	✗	✓	✗	✗	✓
Zeager et al. [79]	✗	✗	✓	✗	✓	✗	✓	✗
Shokri et al. [80]	✓	✗	✗	✗	✓	✗	✗	✓
Biggio et al. [81]	✗	✗	✓	✗	✓	✗	✗	✓
Sharif et al. [82]	✗	✓	✓	✗	✓	✗	✗	✓

Table 10: Comparative analysis of Other Applications at Preparation Phase

	Attack Specif.		Attack Mode		Attack Type	
Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion
Uther et al. [76]	✗	✓	✗	✓	✗	✓
Guillory et al. [77]	✓	✗	✗	✓	✓	✗
Papernot et al. [18]	✗	✓	✗	✓	✗	✓
Alfeld et al. [78]	✗	✓	✓	✗	✓	✗
Zeager et al. [79]	✗	✓	✗	✓	✗	✓
Shokri et al. [80]	✗	✓	✗	✓	✗	✓
Biggio et al. [81]	✓	✗	✗	✓	✓	✗
Sharif et al. [82]	✓	✓	✗	✓	✗	✓

Table 11: Comparative analysis of Other Applications at Manifestation Phase

	Eval. Approach			Perform. Impact		
Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.
Uther et al. [76]	✓	✗	✗	✗	✓	✗	✗
Guillory et al. [77]	✓	✗	✓	✗	✗	✗	✓
Papernot et al. [18]	✓	✗	✓	✗	✗	✓	✗
Alfeld et al. [78]	✗	✗	✓	✗	✗	✓	✗
Zeager et al. [79]	✗	✗	✓	✗	✓	✗	✗
Shokri et al. [80]	✗	✗	✓	✓	✗	✗	✗
Biggio et al. [81]	✗	✗	✓	✓	✗	✗	✗
Sharif et al. [82]	✓	✗	✓	✗	✗	✓	✗

Table 12: Comparative analysis of Other Applications' Attack Evaluation

3.4.4. Autoregressive Forecasting Models

Alfeld et al. [78] have presented a mathematical framework of an attack method against autoregressive forecasting models (*classification*). In such models, an attacker aims to augment the initial values (*poisoning* and *indiscriminate*) so that the forecast ones, computed by an auto-regressive forecasting model, will be as close as possible to the goal of minimizing the difference of the forecast and the desired value. The graybox attack aims to increase *both false positives and false negatives* of the forecasting model.

3.4.5. Game Theory

Zeager et al. [79] have studied the problem of attacks in credit card fraud detection systems that classify (*classification*) charges as fraudulent or not, and they modeled it as a game. The attacker by launching whitebox attacks aims at modifying the attributes of the transactions in order to make fraudulent charges and get undetected (increase *false negatives*). The adversarial classifier

is retrained in multiple rounds of the game using a logistic regression model. The attacker then chooses the subset of fraudulent transactions that utilize the best strategy (*evasion* attack and *indiscriminate* specificity).

3.4.6. Deep Learning

Shokri et al. [80] shed light on a very important area of machine learning, which focuses on how *classification* machine learning models leak information about the individual data records on which they were trained. They demonstrated an *inference membership* attack, where the adversary, launches black-box queries to an ML model, aiming to verify whether some data records were used to train the model (*evasion* attack). In more detail the attacker trains an attack model to distinguish whether a sample was used in the training of the target or not. In order to train the attack model, the adversary trains first a number of *shadow models* which imitate the target's behaviour (increase *false positives* rate). Since the attacker generates a number of shadow models that are distributed similarly to the target model's training dataset, this is an *indiscriminate* attack.

In the context of biometric face verification systems (*classification*), Biggio et al. [81] proposed two face recognition attacks, where they demonstrated how to poison (*poisoning* attack) biometric systems that learn from examples (both *targeted* attacks). In these cases, the attacker aims to impersonate a targeted client (increasing *false positive* rate), by launching whitebox attacks against system.

Sharif et al. [82] demonstrated techniques against facial biometric systems. More precisely they focused on inconspicuous and physically realizable attacks that generate accessories in the form of glass frames to deceive the classifier (*classification*). This technique was studied to both misclassify samples to any other than the correct class, increasing *both false positives and false negatives* and impersonate others using the eyeglass frames. The adversaries in their experiments launch graybox or whitebox attacks against the classifier by observing

its outputs (*evasion* attack) for some inputs. They investigated two categories of attacks; impersonation and dodging. The former is *indiscriminate*, as the attacker aims to have any face recognized as a specific other face, while the latter is a *targeted* attack, since the attacker seeks to have her face misidentified as any other arbitrary face.

3.5. Multipurpose

3.5.1. Naive Bayes - Principal Component Analysis

Huang et al. [10] were one of the first teams to shed light adversarial attacks for machine learning. Besides proposing a taxonomy, they also presented attacks in the context of spam detection and network anomaly detection. They present the adversaries working against *classification* and *clustering* mechanisms by launching blackbox to whitebox attacks. Their spam filtering attacks can be either be *evasion* attacks that increase *false negatives* or *poisoning* attacks that affect *both false positives and false negatives* depending on if they want to evade the detector or cause denial of service. They can also be either *targeted* or *indiscriminate* in terms of attack specificity. In the case of network anomaly detection the attacks are mostly*poisoning* as the adversary poisons the training dataset to evade detection (increase *false negatives*).They are also *indiscriminate* adversaries who cause *many* false negatives. The authors model secure learning systems as a game between an attacker and a defender, in which the attacker manipulates data to mis-train or evade a learning algorithm chosen by the defender to thwart the attacker's objective.

3.5.2. Support Vector Machine (SVM)

Biggio et al. [32] make use of the gradient ascent method, which is a crude algorithmic procedure to perform a graybox poisoning attack against the *Support Vector Machine (SVM)* (*classification*) algorithm by finding a specific point (*targeted*) whose addition to the training dataset maximally decreases the classification accuracy. The *poisoning* methodology involves specially crafted attack

points injected into the training data. The attack affects *both false positives and false negatives* as it increases the classifier's error rate.

Zhou et al. [83] studied two evasion (*evasion* attack) attack models in the context of spam email detection systems and credit card fraud detection systems (*classification*) where an adversary launches graybox to whitebox attacks, is trying to avoid detection (increase *false negatives*): in the *free-range* attacks the adversary is able to move the data anywhere in the feature space while on the *restrained* attacks he sets bounds on the displacement of the features. The latter attack model proved the existence of a trade-off between disguising malicious and retaining their malicious utility. In the restrained attack, the adversary alters one single instance (*targeted*) while in the free-range attack the adversary aims to perturb the entire set of instances (*indiscriminate*).

Demontis et al. [84] focus on the vulnerability of linear classifiers (*classification*) in the context of handwritten digit classification, spam filtering and pdf malware detection. They investigated *evasion* attacks. According to their threat model an adversary launches graybox to whitebox attacks by modifying specific malicious samples (*targeted*) during the test phase, having as goal to misclassify them as benign (increase *false negatives*).

Zhang et al. [85] performed experiments on spam detection and PDF malware detection systems and proposed two algorithms for feature selection against adversaries. Under adversarial settings, the use of reduced feature sets resulted in improved computational complexity and in more robust classification against graybox to whitebox *evasion* attacks that try to avoid detection (increase *false negatives*). Both algorithms aim at maximizing the generalization capability of the linear classifier (*classification*) and the robustness to evasion attacks, both consisting the objective function of the optimization problem (*indiscriminate*).

3.5.3. Support Vector Machine (SVM) - Logistic Regression

The same author and his team [86] assumed that adversaries, launch whitebox attacks against a classifier in a variety of applications: spam email detection

(*indiscriminate*), biometric authentication (*targeted*) and network intrusion detection (*indiscriminate*), can launch *evasion* attacks depending on an arms race game and affecting the distribution of training and testing data separately. The final goal of those attacks is to avoid detection of their activities (*false negatives* rate) by the classifier (*classification*).

Mei et al. [87] presented an efficient solution for a broad class of *poisoning* attacks against spam email detection systems and wine quality modelling systems. They managed to find an optimal training-set and attack a broad family of machine learners by solving a bilevel optimization problem (*indiscriminate*). In training-set attacks, an attacker contaminates the training data so that a specific learning algorithm would produce a model profitable to the attacker. Under natural assumptions (differentiable space and convex objective function) they reduced the problem to a single level optimization. By using the Karush-Kuhn-Tucker (KKT) conditions, they derived three concrete cases of attacks against *classification* (logistic regression, SVMs and linear regression) by only being allowed to change the features (*whitebox* attacks) during the training phase. Their assumption was that the adversary is trying to bypass classifier's detection (increase *false negatives*).

3.5.4. Support Vector Machine (SVM) - Multiple Classifier System (MCS)

In continuation of their work, the same team [88] created *one and a half class* classifier (*classification*), in order to achieve a good trade-off between classification accuracy and security. They experimented with real world data in spam detection and PDF malware detection to test their assumption. According to their attack model an attacker can launch graybox to whitebox *evasion* attacks against the classifier by modifying malicious data at test time in order to have malicious samples misclassified as legitimate (increasing *false negatives*). Authors assume that the attacker manipulates an initial malicious sample, characterizing this as a *targeted* attack, to minimize the classifier's discriminant function.

	Attacker Knowl.			Learning			Adv.. Strat.			Attacker Knowl.			Learning			Adv. Strat.	
Ref.	Blackbox	Graybox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.	Ref.	Blackbox	Whitebox	Whitebox	Clustering	Classification	Hybrid	Gam. Th.	No Gam. Th.
Huang et al. [10]	✓	✓	✓	✓	✓	✗	✗	✓	Biggio et al. [32]	✗	✓	✗	✗	✓	✗	✗	✓
Biggio et al. [86]	✗	✗	✓	✗	✓	✗	✗	✓	Biggio et al. [88]	✗	✓	✓	✗	✓	✗	✗	✓
Liu et al. [89]	✗	✓	✗	✗	✓	✗	✓	✗	Bulo et al. [90]	✗	✓	✗	✗	✓	✗	✓	✗
Gonzalez et al. [91]	✗	✓	✓	✗	✓	✗	✗	✓	Alpcan et al. [92]	✗	✗	✓	✗	✓	✗	✓	✗
Demontis et al. [84]	✗	✓	✓	✗	✓	✗	✗	✓	Gonzalez et al. [93]	✗	✓	✓	✗	✓	✗	✗	✓
Li et al. [94]	✗	✗	✓	✓	✗	✗	✓	✗	Louppe et al. [95]	✗	✓	✓	✓	✗	✗	✗	✓
Wang et al. [20]	✓	✗	✓	✗	✓	✗	✗	✓	Bhagoji et al. [96]	✗	✗	✓	✗	✓	✗	✗	✓
Cisse et al. [97]	✗	✓	✗	✗	✓	✗	✗	✓	Mei et al. [87]	✗	✗	✓	✗	✓	✗	✗	✓
Zhou et al. [83]	✗	✓	✓	✗	✓	✗	✗	✓	Zhang et al. [85]	✗	✓	✓	✗	✓	✗	✗	✓

Table 13: Comparative analysis of Multipurpose at Preparation Phase

3.5.5. Game Theory

Liu et al. in [89] in the context of handwritten digit recognition and spam detection (*classification*) modeled the interaction of a data miner and an adversary using a one-step game-theoretical model, where the adversary aims to minimize both the difference between distributions of positive and negative classes and the adversarial movement itself (increase *false negatives*). They provided a more computationally efficient algorithm to find the NE of a binary classification by solving one maximin optimization problem only (*indiscriminate*). The adversary modifies the feature vectors (graybox *evasion* attack) in order to shift them during the test phase.

Bulo et al. [90] extended the work of Bruckner et al. [48] on *static prediction games* by introducing randomization to the previous model, applied on a variety of domains (spam email detection system, handwritten digit recognition, PDF malware detection). Their experimental results on various applications but mainly on handwritten recognition indicate that the classifiers (*classification*)

can be learned by the adversary (graybox attacks) when he deviates from his hypothesized action by the learner. However in order for this to happen, the adversary needs to apply several modifications to the malicious samples at test time (*evasion* attack). In addition to that, the threat model includes attacks in which malicious samples are manipulated at test time to evade detection increasing *false negatives*. Similar to [48], the adversary performs *indiscriminate* attacks.

Alpcan et al. [92] use game theory to provide adversarial machine learning strategies based on linear Support Vector Machines (SVMs), which leads to a large-scale game formulation. In a variety of domains (communication networks, smart electricity grids, and cyber-physical systems), a Support Vector Machine (SVM) is considered as a binary classifier (*classification*). Their threat model assumes whitebox distortions of any training data points (*poisoning* attack and *indiscriminate*), which are attacks launched by an attacker who injects data to increase *false negatives* aiming misclassification of his data.

3.5.6. Deep Learning

Munoz et al. [91] proposed a poisoning attack model in order to maximize the misclassification (*both false positives and false negatives*) performance of various learning algorithms. In contrast to previous works that studied poisoning attacks solely against binary learners, the authors are the first to study this problem against multiclass classification (*classification*) settings extending previously studied models. As in graybox to whitebox *poisoning* attacks, the attacker controls a fraction of the training data and the attacker's knowledge varies. This paper investigates both *targeted* and *indiscriminate* attacks.

Gonzalez et al. [93], proposed a novel poisoning algorithm based on the idea of back-gradient optimization, i.e., to compute the gradient of interest through automatic differentiation, while also reversing the learning procedure to drastically reduce the attack complexity in *classification*. Their threat model includes an adversary launches either *targeted* or *indiscriminate* graybox to whitebox

attacks with either *poisoning* or*evasion* attacks. The adversary's final goal is to increase either *false negatives* or *both false positives and false negatives*.

Wang et al. [20] proposed an adversary resistant model in order to construct robut DNNs. Their methodology, Random Feature Nullification (RFN), randomly nullifies features within the data vectors. Experiments were conducted against malware and image recognition samples on MNIST and CIFAR-10 datasets. The experiments includedwhitebox and blackbox *poisoning* attacks during the training phase, aiming to make the classifier classify a benign application as malicious (increase *false positives*). The experiments proved that the classification accuracy (*classification*) decreases slightly in all the cases when the nullification rate increases while on contrary the resistance increases significantly. As the adversarial perturbation is generated by computing the derivative of the DNN's cost function with respect to the input samples, this is characterized as an *indiscriminate* attack.

Cisse et al. [97] used MNIST, CIFAR-10, CIFAR-100 and Street View House Numbers as datasets and they proposed an efficient algorithm for data augmentation during training of DNNs that when combined with the use of Parseval networks results in robustness of a system against adversarial noise. Parseval networks are used as a multi-layer regularisation method to reduce the sensitivity of DNNs against perturbed noisy samples. The authors performed experiments where adversaries can poison training (graybox *poisoning* attack) of an image classifier (*classification*) in order to avoid detection (increase *false negatives*). The investigated adversarial model assumes that the attacker causes a small perturbation of the input pattern with the goal to remain stealthy; thus we ascertain that this is a *targeted* attack.

3.5.7. Generative Adversarial Networks (GAN)

Li et al. [94] focused on Generative Adversarial Networks (GANs) considering a variety of applications such as image translation, speech synthesis and robot trajectory prediction. GANs are a part of *clustering* methodologies. The

authors proposed that an adversary who participates in a two player game, is able to launch a whitebox *poisoning* attack, aiming to cause *clustering accuracy reduction* of the system by generating samples that are similar to the data distribution (*indiscriminate*). Louppe and Cranmer [95] also working in GANs, focused on computer simulations and on how they could use data generation to help a variety of different scientific fields (e.g., particle physics, epidemiology, and population genetics). They proposed a likelihood-free inference algorithm for fitting a non-differentiable generative model incorporating ideas from empirical Bayes and variational inference. Their novelty is that they use a domain specific simulator instead of a differentiable generative network. Their perception of an adversary against
textcolorblack*clustering* is that he is able to launch graybox to whitebox *poisoning* attacks to inflict *clustering accuracy reduction* of the attacked system. The proposed Adversarial variational optimization follows the *indiscriminate* pattern of attacks.

3.5.8. Support Vector Machine (SVM) - Deep Learning

Bhagoji et al. [96] focused on image recognition and human activity recognition to use linear transformation as data defence against *evasion* attacks. According to their threat model the attacker has perfect knowledge of the classifier (*classification*) as he can launch *whitebox* attacks. Their threat models covers two cases: Targeted and Untargeted whether the attackers aims at classifying samples as part of a specific class or misclassifying all samples, respectively (*targeted* and *indiscriminate*). Also, the attacks increase *both false positives and false negatives*.

	Attack Specif.		Attack Mode		Attack Type			Attack Specif.		Attack Mode		Attack Type	
Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion	Ref.	Targeted	Indiscriminate	Colluding	Non-colluding	Poisoning	Evasion
Huang et al. [10]	✓	✓	✗	✓	✓	✗	Biggio et al. [32]	✓	✗	✗	✓	✓	✗
Biggio et al. [86]	✓	✓	✗	✓	✗	✓	Biggio et al. [88]	✓	✗	✗	✓	✗	✓
Liu et al. [89]	✗	✓	✗	✓	✗	✓	Bulo et al. [90]	✗	✓	✗	✓	✗	✓
Gonzalez et al. [91]	✓	✓	✗	✓	✓	✗	Alpcan et al. [92]	✗	✓	✗	✓	✓	✗
Demontis et al. [84]	✓	✗	✗	✓	✗	✓	Gonzalez et al. [93]	✓	✗	✗	✓	✗	✓
Li et al. [94]	✗	✓	✗	✓	✓	✗	Louppe et al. [95]	✗	✓	✗	✓	✓	✗
Wang et al. [20]	✗	✓	✗	✓	✓	✗	Bhagoji et al. [96]	✓	✓	✗	✓	✗	✓
Cisse et al. [97]	✓	✗	✓	✗	✓	✗	Mei et al. [87]	✗	✓	✗	✓	✓	✗
Zhou et al. [83]	✓	✓	✗	✓	✗	✓	Zhang et al. [85]	✗	✓	✗	✓	✗	✓

Table 14: Comparative analysis of Multipurpose at Manifestation Phase

	Eval. Appr.			Perf. Impact					Eval. Appr.			Perf. Impact			
Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.	Ref.	Analytical	Simulation	Experimental	False Positives	False Negatives	Both FP and FN	Clust. Accur. Red.
Huang et al. [10]	✗	✗	✓	✗	✓	✓	✗	Biggio et al. [32]	✗	✗	✓	✗	✗	✓	✗
Biggio et al. [86]	✓	✗	✓	✗	✓	✗	✗	Biggio et al. [88]	✗	✗	✓	✗	✓	✗	✗
Liu et al. [89]	✓	✓	✓	✗	✓	✗	✗	Bulo et al. [90]	✓	✗	✓	✗	✓	✗	✗
Gonzalez et al. [91]	✗	✗	✓	✗	✗	✓	✗	Alpcan et al. [92]	✓	✗	✓	✗	✓	✗	✗
Demontis et al. [84]	✓	✗	✓	✗	✓	✗	✗	Gonzalez et al. [93]	✓	✗	✓	✗	✓	✓	✗
Li et al. [94]	✓	✗	✓	✗	✗	✗	✓	Louppe et al. [95]	✓	✗	✗	✗	✗	✗	✓
Wang et al. [20]	✗	✗	✓	✓	✗	✗	✗	Bhagoji et al. [96]	✗	✗	✓	✗	✗	✓	✗
Cisse et al. [97]	✓	✗	✓	✗	✓	✗	✗	Mei et al. [87]	✗	✗	✓	✗	✓	✗	✗
Zhou et al. [83]	✗	✗	✓	✗	✓	✗	✗	Zhang et al. [85]	✗	✗	✓	✗	✓	✗	✗

Table 15: Comparative analysis of Multipurpose's Attack Evaluation

4. Discussion and Conclusions

4.1. Distribution of papers per taxonomy phase

Figures 4-5 present the volume of papers for the different features of each taxonomy phase, while Figure 5 illustrates the volume of papers for the evaluation approach and the performance impact after the manifestation phase. The motivation behind this is to derive trends about the degree that each feature type has been investigated. This may assist with decisions regarding future work and the identification of less investigated feature types. It is worth mentioning that these numbers (i.e., number of papers per feature) will not sum up to the total number of papers (i.e., 66) because many papers are investigating more than one of the identified feature types.

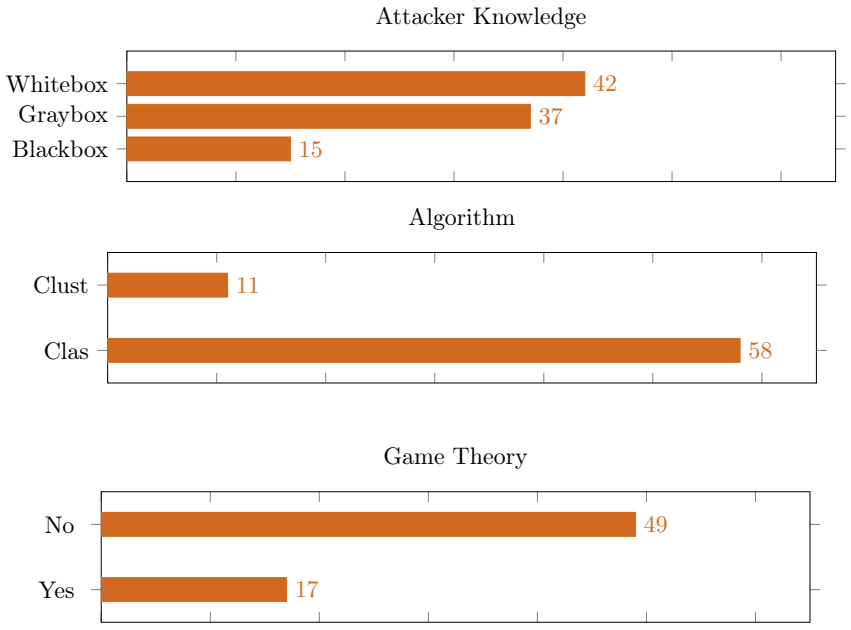

Figure 4: Volume of papers for each *preparation phase* feature.

In Fig. 4, we observe that 64% of the papers investigate the cases of *whitebox* attacks. This is expected because it is intuitive for authors to first investigate

attacks that have knowledge of the system, training data and machine learning algorithms. These can be insider threats. To investigate different ways the attacker targets machine learning system and to make their defense solutions, when this is provided, authors in approximately 11% of the papers investigate all three different levels of attacker knowledge. Around 56% of the papers investigate blackbox attacks, which assumes that one of the parameters (e.g., training data) is not known.

The above percentages in terms of papers that do not study exclusively attackers who launch *whitebox* attacks, are supported by the findings of our survey regarding the number of papers that investigate *Evasion* Attacks (*manifestation* phase). This is 52% of all papers surveyed and on many occasions are studied in conjunction with the *Poisoning* Attacks which alone appear in56% of all papers. The reason being that adversaries leverage *evasion* attacks to escalate their knowledge level.

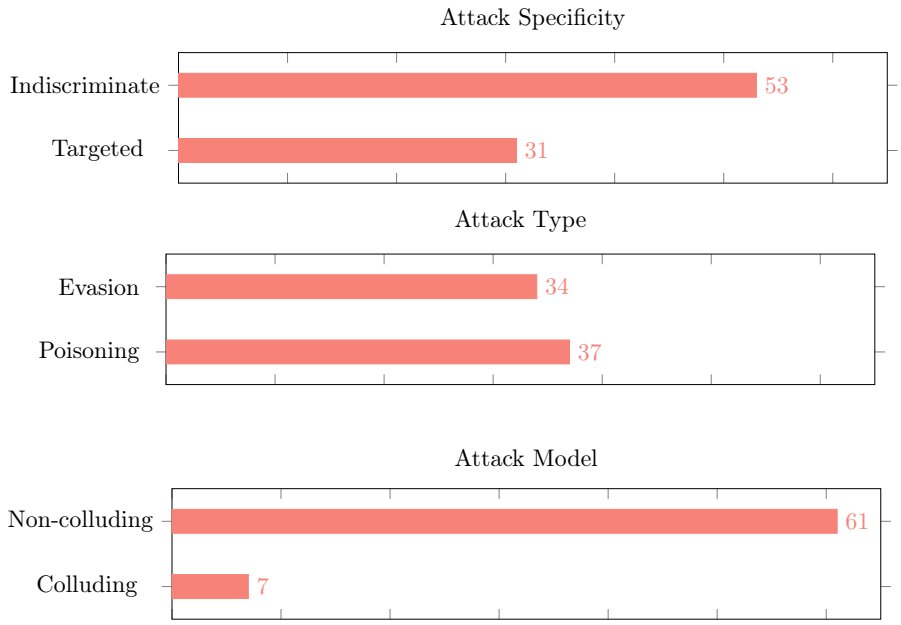

Figure 5: Volume of papers for each *manifestation phase* feature.

Regarding the remaining preparation phase features, attacks on *classification* are studied in 88% of the papers, while for *clustering*, this drops to only 17%. Furthermore, a surprisingly high number of papers (26%) study adversarial machine learning using Game Theory. Given the strategic interactions between defender (machine learning system) and attacker, strategic choices provided by game theory in the form of game equilibria appear as an attractive way to optimize player's strategies in various different types of games. It is worth mentioning that such approaches can be used to design optimal defences with the assumption that the defender solves a zero-sum game thus preventing maximum damages incurred by the adversary. Also, we observe that all papers that have not proposed game-theoretic models account for 74% of all papers showing that there aren't papers that study both types of this feature.

With regards to the manifestation phase, apart from Attack Influence that we have discussed earlier, 80% of the papers study Indiscriminate Attacks, where the adversary attacks a very general class of samples. The high percent can be justified by the benefit of making an attack more likely to be successful due to the wide range of "targets" (i.e., samples) aimed by the adversary. When combining different statistics, we also observe that papers in which the attacker has exclusively Zero Knowledge, the attack is always indiscriminate meaning that for a targeted attack to take place some prior knowledge must have been acquired. Almost half of all papers (47%) study Targeted Attacks.

It is also worth highlighting that the vast majority of papers (92%) assume non-colluding attackers, while only 11% introduces the notion of collaboration among adversaries to manifest the attack. Interestingly, only one ([46]) uses game theory to model interactions between a learner and a collection of adversaries whose objectives are inferred from training data. This work paves the way for future work in the field towards inferring game strategies to defend against collaborative attackers.

With regards to the evaluation approach, 95% have conducted experiments with some realistic datasets to assess the performance of their methodology. This shows that they do not analyze adversarial machine learning in a purely the-

oretic way but conduct experiments to measure the required parameters (false positives, false negatives, clustering accuracy detection). Approximately 38% of the papers assess their proposed framework both analytically and experimentally with only 3 papers to restrict their assessment using analytical methods. It is worth noting that [89] is the only paper that undertakes simulations, and this in addition to experimental and analytical evaluation. Finally, 41% of all papers measure both False Positives and False Negatives (FNs) whilst 48% measure FNs alone. Only 9% and 6% of the papers measure the Clustering Accuracy Reduction and FPs respectively.

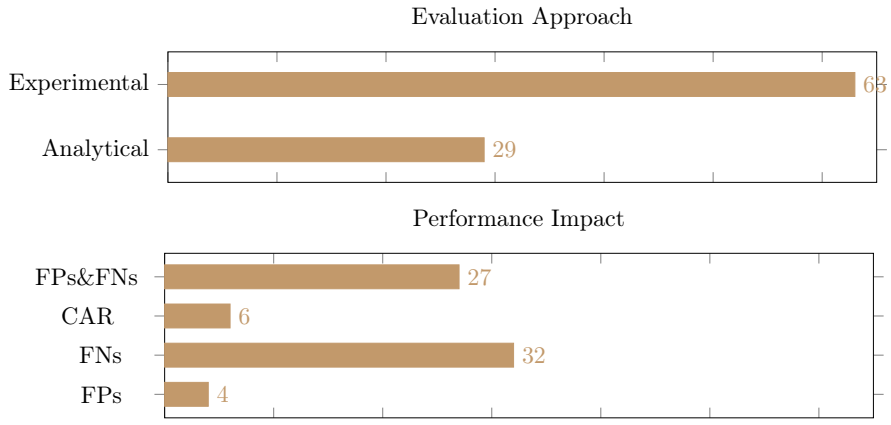

Figure 6: Volume of papers for each feature after the manifestation phase, where FPs: False Positives, FNS: False Negatives, CAR: Clustering Accuracy Reduction.

With regards to the usage of deep learning in the application categories (Figure 7) we have studied we cn identify that Visual Recognition has the most frequent usage of deep learning (86% of the papers of its category) with Other Applications ranking second (50% of the papers of this category) and Multi-purpose ranking third (39% of the papers of this category). Deep learning did not appear in Spam Filtering, while it made only one appearance in Intrusion Detection category. This makes perfect sense as deep learning applications are mostly used in visual recognition systems and biometric recognition systems with the most popular being face recognition systems. Spam Filtering and

Intrusion Detection systems are build mostly on top of linear methodologies. In total 45% of our pool of papers was occupied with papers containing deep learning methodologies, which highlights the significance of this specific machine learning approach when it comes to adversaries.

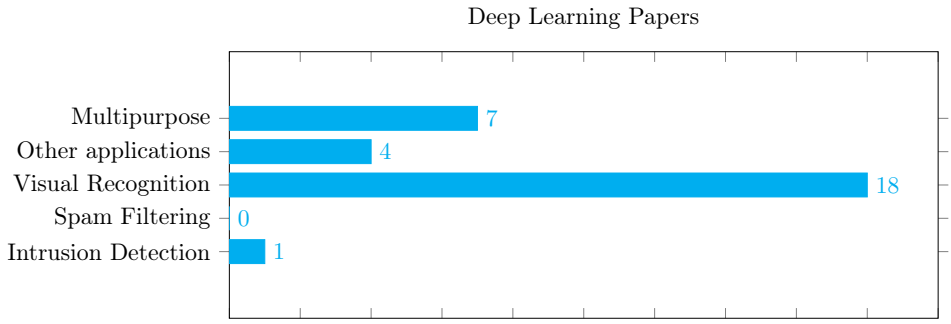

Figure 7: Volume of Deep Learning papers for each application domain

4.2. Open problems

Machine learning defensive mechanisms are evolving as new attacks are introduced in the scientific community and new application areas are involved. Throughout the papers that we have surveyed and classified using our taxonomy, a number of open problems have drawn our attention as areas for future research.

Creating immunity to perturbations. A defensive approach that is popular in the literature is to train a system using modified data, so as to increase its robustness against perturbations of input. This is a countermeasure that is useful as long as the adversary keeps using the same algorithm for the perturbations. In Chen et al. [40] retraining the classifier with noise resulted in the increase of average prediction confidence but not necessarily of the accuracy. As soon as there is any change in the algorithm, the accuracy of any detection system drops even in the physical world [66]. Hayes et al. [58] have created an attack that can overcome a wide range of defenses, and in [62] they found out that models trained in an adversarial manner are still vulnerable to Universal

Adversarial Perturbations trained against the defended model. For this reason, Papernot et al. [64] concluded that defending against finite perturbations is a more promising avenue for future work than defending against infinitesimal perturbations.

Multiple adversaries. Dalvi et al. [30] point out that the challenge is substantially different when a machine learning system has to face multiple adversaries instead of one. In the majority of the literature, only a small number of papers have investigated models with multiple adversaries who may collaborate towards a common goal. Consequently, colluding attacks within Adversarial Machine Learning is a challenge that can be investigated further.

Randomized adversarial approaches. According to Barreno et al. [29], randomization of the distribution of the training data can increase the adversary's work, but it will also increase the learner's initial error rate before randomization. Determining the right amount of randomization is still an open problem. Also, the same author [35] raised the issue of reflecting the lack of a mechanism that can measure the value of the leaked information from a learning system to an attacker, thus quantifying the risk associated with various side channel attacks that exploit leaked information.

Digital forensics for adversarial machine learning. Almost all existing research on addressing adversarial machine learning is focused on preventing and protecting systems against it. However, given its impact in physical space [65] and potential in assisting crime, it is important to also consider how to forensically analyze such attacks while adhering to the principles of digital evidence. The breadth and depth of machine learning knowledge required to spot adversarial behaviour may be a barrier to many digital forensic professionals. Here, it is likely that experience from multimedia forensics can help considerably, as attempted already in [68]. What can be extremely useful is that researchers develop methodologies, ideally supported by new software toolkits, allowing digital forensic professionals to spot and analyze signs of purposeful adversarial behaviour where the failure of a machine learning process causes damage or assists a crime.

5. Conclusions

The wide adoption of machine learning techniques in almost every aspect of modern society has made it a very popular area of scientific research in terms of both attack and defense mechanisms. We have proposed a comprehensive taxonomy of the characteristics of adversarial machine learning approaches using existing well-cited terminology. Based on the latter, we systematically reviewed the landscape of adversarial machine learning approaches, paying special attention to the application area. In addition to that, we pointed out existing open problems that require the attention of the scientific community.

In the future we plan to address the limitations of the previous approaches and especially the vulnerability of machine learning against perturbations creating a framework, that will digest each newly introduced attack pattern and adopt accordingly. Additionally, we envisage the creation of a unified infrastructure which will acquire information for attacks against machine learning through different application probes, thus creating a strong knowledge base which will help augment the defences against adversarial attempts.

References

[1] S. G. Finlayson, H. W. Chung, I. S. Kohane, A. L. Beam, Adversarial attacks against medical deep learning systems, arXiv preprint arXiv:1804.05296 (2018).

[2] N. Pitropakis, A. Pikrakis, C. Lambrinoudakis, Behaviour reflects personality: detecting co-residence attacks on xen-based cloud environments, International Journal of Information Security 14 (2015) 299–305.

[3] J. Park, M.-H. Kim, S. Choi, I. S. Kweon, D.-G. Choi, Fraud detection with multi-modal attention and correspondence learning, in: 2019 International Conference on Electronics, Information, and Communication (ICEIC), IEEE, 2019, pp. 1–7.

[4] G. Loukas, E. Karapistoli, E. Panaousis, P. Sarigiannidis, A. Bezemskij, T. Vuong, A taxonomy and survey of cyber-physical intrusion detection approaches for vehicles, Ad Hoc Networks 84 (2019) 124–147.

[5] T. Giannetsos, T. Dimitriou, N. R. Prasad, People-centric sensing in assistive healthcare: Privacy challenges and directions, Security and Communication Networks 4 (2011) 1295–1307.

[6] B. D. Rouani, M. Samragh, T. Javidi, F. Koushanfar, Safe machine learning and defeating adversarial attacks, IEEE Security & Privacy 17 (2019) 31–38.

[7] S. Gisdakis, M. Laganà, T. Giannetsos, P. Papadimitratos, Serosa: Service oriented security architecture for vehicular communications, in: Vehicular Networking Conference (VNC), 2013 IEEE, IEEE, 2013, pp. 111–118.

[8] Y. Liu, Y. Xie, A. Srivastava, Neural trojans, CoRR abs/1710.00942 (2017).

[9] N. Baracaldo, B. Chen, H. Ludwig, A. Safavi, R. Zhang, Detecting poisoning attacks on machine learning in iot environments, in: 2018 IEEE International Congress on Internet of Things (ICIOT), 2018, pp. 57–64.

[10] L. Huang, A. D. Joseph, B. Nelson, B. I. Rubinstein, J. Tygar, Adversarial machine learning, in: Proceedings of the 4th ACM workshop on Security and artificial intelligence, ACM, 2011, pp. 43–58.

[11] R. Huang, B. Xu, D. Schuurmans, C. Szepesvári, Learning with a strong adversary, arXiv preprint arXiv:1511.03034 (2015).

[12] Y. Abbasi, P. L. Bartlett, V. Kanade, Y. Seldin, C. Szepesvári, Online learning in markov decision processes with adversarially chosen transition probability distributions, in: Advances in neural information processing systems, 2013, pp. 2508–2516.

[13] G. Neu, A. Gyorgy, C. Szepesvári, The adversarial stochastic shortest path problem with unknown transition probabilities, in: Artificial Intelligence and Statistics, 2012, pp. 805–813.

[14] M. Brückner, T. Scheffer, Nash equilibria of static prediction games, in: Advances in neural information processing systems, 2009, pp. 171–179.

[15] U. Shaham, Y. Yamada, S. Negahban, Understanding adversarial training: Increasing local stability of neural nets through robust optimization, arXiv preprint arXiv:1511.05432 (2015).

[16] A. Nguyen, J. Yosinski, J. Clune, Deep neural networks are easily fooled: High confidence predictions for unrecognizable images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 427–436.

[17] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, arXiv preprint arXiv:1312.6199 (2013).

[18] N. Papernot, P. McDaniel, A. Swami, R. Harang, Crafting adversarial input sequences for recurrent neural networks, in: Military Communications Conference, MILCOM 2016-2016 IEEE, IEEE, 2016, pp. 49–54.

[19] Y. Zhao, I. Shumailov, R. Mullins, R. Anderson, To compress or not to compress: Understanding the interactions between adversarial attacks and neural network compression, arXiv preprint arXiv:1810.00208 (2018).

[20] Q. Wang, W. Guo, K. Zhang, A. G. Ororbia II, X. Xing, X. Liu, C. L. Giles, Adversary resistant deep neural networks with an application to malware detection, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 1145–1153.

[21] I. Shumailov, Y. Zhao, R. Mullins, R. Anderson, The taboo trap: Behavioural detection of adversarial samples, arXiv preprint arXiv:1811.07375 (2018).

[22] Y. Zhou, M. Kantarcioglu, B. Xi, A survey of game theoretic approach for adversarial machine learning, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery (2018).

[23] N. Akhtar, A. Mian, Threat of adversarial attacks on deep learning in computer vision: A survey, Access (2018).

[24] J. Zhang, X. Jiang, Adversarial examples: Opportunities and challenges, arXiv preprint arXiv:1809.04790 (2018).

[25] V. Duddu, A survey of adversarial machine learning in cyber warfare., Defence Science Journal 68 (2018).

[26] L. Sun, M. Tan, Z. Zhou, A survey of practical adversarial example attacks, Cybersecurity 1 (2018) 9.

[27] B. Biggio, F. Roli, Wild patterns: Ten years after the rise of adversarial machine learning, Pattern Recognition 84 (2018) 317–331.

[28] X. Yuan, P. He, Q. Zhu, X. Li, Adversarial examples: Attacks and defenses for deep learning, IEEE transactions on neural networks and learning systems (2019).

[29] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, J. D. Tygar, Can machine learning be secure?, in: Proceedings of the 2006 ACM Symposium on Information, computer and communications security, ACM, 2006, pp. 16–25.

[30] N. Dalvi, P. Domingos, S. Sanghai, D. Verma, et al., Adversarial classification, in: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2004, pp. 99–108.

[31] N. Banerjee, T. Giannetsos, E. Panaousis, C. C. Took, Unsupervised learning for trustworthy IoT, CoRR abs/1805.10401 (2018).

[32] B. Biggio, B. Nelson, P. Laskov, Poisoning attacks against support vector machines, arXiv preprint arXiv:1206.6389 (2012).

[33] B. Chen, W. Carvalho, N. Baracaldo, H. Ludwig, B. Edwards, T. Lee, I. Molloy, B. Srivastava, Detecting backdoor attacks on deep neural networks by activation clustering, arXiv preprint arXiv:1811.03728 (2018).

[34] Y. Vorobeychik, Adversarial ai., in: IJCAI, 2016, pp. 4094–4099.

[35] M. Barreno, B. Nelson, A. D. Joseph, J. Tygar, The security of machine learning, Machine Learning 81 (2010) 121–148.

[36] Y. LeCun, The mnist database of handwritten digits, http://yann. lecun. com/exdb/mnist/ (1998).

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, Ieee, 2009, pp. 248–255.

[38] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, F. Roli, Evasion attacks against machine learning at test time, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2013, pp. 387–402.

[39] H. Xiao, B. Biggio, G. Brown, G. Fumera, C. Eckert, F. Roli, Is feature selection secure against training data poisoning?, in: International Conference on Machine Learning, 2015, pp. 1689–1698.

[40] Y. Chen, Y. Nadji, A. Kountouras, F. Monrose, R. Perdisci, M. Antonakakis, N. Vasiloglou, Practical attacks against graph-based clustering, arXiv preprint arXiv:1708.09056 (2017).

[41] B. I. Rubinstein, B. Nelson, L. Huang, A. D. Joseph, S.-h. Lau, S. Rao, N. Taft, J. Tygar, Antidote: understanding and defending against poisoning of anomaly detectors, in: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, ACM, 2009, pp. 1–14.

[42] G. Wang, T. Wang, H. Zheng, B. Y. Zhao, Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers, in: USENIX Security Symposium, 2014, pp. 239–254.

[43] L. Demetrio, B. Biggio, G. Lagorio, F. Roli, A. Armando, Explaining vulnerabilities of deep learning to adversarial malware binaries, arXiv preprint arXiv:1901.03583 (2019).

[44] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, C. K. Nicholas, Malware detection by eating a whole exe, in: Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[45] D. Lowd, C. Meek, Adversarial learning, in: 11th ACM SIGKDD international conference on Knowledge discovery in data mining, ACM, 2005, pp. 641–647.

[46] B. Li, Y. Vorobeychik, Feature cross-substitution in adversarial classification, in: Advances in neural information processing systems, 2014, pp. 2087–2095.

[47] M. Brückner, T. Scheffer, Stackelberg games for adversarial prediction problems, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2011, pp. 547–555.

[48] M. Brückner, C. Kanzow, T. Scheffer, Static prediction games for adversarial learning problems, Journal of Machine Learning Research 13 (2012) 2617–2654.

[49] W. Liu, S. Chawla, A game theoretical model for adversarial learning, in: IEEE International Conference on Data Mining Workshops, IEEE, 2009, pp. 25–30.

[50] R. Zhang, Q. Zhu, Secure and resilient distributed machine learning under adversarial environments, in: Information Fusion (Fusion), 2015 18th International Conference on, IEEE, 2015, pp. 644–651.

[51] M. Großhans, C. Sawade, M. Brückner, T. Scheffer, Bayesian games for adversarial regression problems, in: International Conference on Machine Learning, 2013, pp. 55–63.

[52] B. Nelson, M. Barreno, F. J. Chi, A. D. Joseph, B. I. Rubinstein, U. Saini, C. A. Sutton, J. D. Tygar, K. Xia, Exploiting machine learning to subvert your spam filter., LEET 8 (2008) 1–9.

[53] R. Naveiro, A. Redondo, D. R. Insua, F. Ruggeri, Adversarial classification: An adversarial risk analysis approach, arXiv preprint arXiv:1802.07513 (2018).

[54] H. Xiao, B. Biggio, B. Nelson, H. Xiao, C. Eckert, F. Roli, Support vector machines under adversarial label contamination, Neurocomputing 160 (2015) 53–62.

[55] N. Carlini, D. Wagner, Adversarial examples are not easily detected: Bypassing ten detection methods, arXiv preprint arXiv:1705.07263 (2017).

[56] J. J. Hull, A database for handwritten text recognition research, IEEE Transactions on pattern analysis and machine intelligence 16 (1994) 550–554.

[57] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Technical report, University of Toronto (2009).

[58] J. Hayes, G. Danezis, Machine learning as an adversarial service: Learning black-box adversarial examples, arXiv preprint arXiv:1708.05207 (2017).

[59] B. Biggio, I. Pillai, S. Rota Bulò, D. Ariu, M. Pelillo, F. Roli, Is data clustering in adversarial settings secure?, in: Proceedings of the 2013 ACM workshop on Artificial intelligence and security, ACM, 2013, pp. 87–98.

[60] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, arXiv preprint arXiv:1412.6572 (2014).

[61] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in neural information processing systems, 2014, pp. 2672–2680.

[62] J. Hayes, G. Danezis, Learning universal adversarial perturbations with generative models, in: 2018 IEEE Security and Privacy Workshops (SPW), IEEE, 2018, pp. 43–49.

[63] N. Papernot, P. McDaniel, I. Goodfellow, Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, arXiv preprint arXiv:1605.07277 (2016).

[64] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, A. Swami, Practical black-box attacks against machine learning, in: ACM on Asia Conference on Computer and Communications Security, ACM, 2017, pp. 506–519.

[65] A. Kurakin, I. Goodfellow, S. Bengio, Adversarial machine learning at scale, arXiv preprint arXiv:1611.01236 (2016).

[66] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, D. Song, Robust physical-world attacks on machine learning models, arXiv preprint arXiv:1707.08945 (2017).

[67] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434 (2015).

[68] P. Schöttle, A. Schlögl, C. Pasquini, R. Böhme, Detecting adversarial examples-a lesson from multimedia forensics, arXiv preprint arXiv:1803.03613 (2018).

[69] A. S. Chivukula, W. Liu, Adversarial learning games with deep learning models, in: Neural Networks (IJCNN), 2017 International Joint Conference on, IEEE, 2017, pp. 2758–2767.

[70] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, arXiv preprint arXiv:1706.06083 (2017).

[71] J. T. Springenberg, Unsupervised and semi-supervised learning with categorical generative adversarial networks, arXiv preprint arXiv:1511.06390 (2015).

[72] A. Athalye, N. Carlini, D. Wagner, Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, arXiv preprint arXiv:1802.00420 (2018).

[73] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, J. Li, Boosting adversarial attacks with momentum, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9185–9193.

[74] J. Jeong, S. Kwon, M.-P. Hong, J. Kwak, T. Shon, Adversarial attack-based security vulnerability verification using deep learning library for multimedia video surveillance, Multimedia Tools and Applications (2019) 1–15.

[75] A. Krizhevsky, V. Nair, G. Hinton, The cifar-10 dataset, online: http://www. cs. toronto. edu/kriz/cifar. html 55 (2014).

[76] W. Uther, M. Veloso, Adversarial reinforcement learning, Technical Report, Tech. rep., Carnegie Mellon University. Unpublished, 1997.

[77] A. Guillory, J. A. Bilmes, Simultaneous learning and covering with adversarial noise, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 369–376.

[78] S. Alfeld, X. Zhu, P. Barford, Data poisoning attacks against autoregressive models., in: AAAI, 2016, pp. 1452–1458.

[79] M. F. Zeager, A. Sridhar, N. Fogal, S. Adams, D. E. Brown, P. A. Beling, Adversarial learning in credit card fraud detection, in: Systems and Information Engineering Design Symposium, IEEE, 2017, pp. 112–116.

[80] R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership inference attacks against machine learning models, in: Security and Privacy (SP), 2017 IEEE Symposium on, IEEE, 2017, pp. 3–18.

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

[82] M. Sharif, S. Bhagavatula, L. Bauer, M. K. Reiter, Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition, in: ACM SIGSAC Conference on Computer and Communications Security, ACM, 2016, pp. 1528–1540.

[83] Y. Zhou, M. Kantarcioglu, B. Thuraisingham, B. Xi, Adversarial support vector machine learning, in: 18th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2012, pp. 1059–1067.

[84] A. Demontis, P. Russu, B. Biggio, G. Fumera, F. Roli, On security and sparsity of linear classifiers for adversarial settings, in: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Springer, 2016, pp. 322–332.

[85] F. Zhang, P. P. Chan, B. Biggio, D. S. Yeung, F. Roli, Adversarial feature selection against evasion attacks, IEEE transactions on cybernetics 46 (2016) 766–777.

[86] B. Biggio, G. Fumera, F. Roli, Security evaluation of pattern classifiers under attack, IEEE transactions on knowledge and data engineering 26 (2014) 984–996.

[87] S. Mei, X. Zhu, Using machine teaching to identify optimal training-set attacks on machine learners, in: AAAI, 2015, pp. 2871–2877.

[88] B. Biggio, I. Corona, Z.-M. He, P. P. Chan, G. Giacinto, D. S. Yeung, F. Roli, One-and-a-half-class multiple classifier systems for secure learning against evasion attacks at test time, in: International Workshop on Multiple Classifier Systems, Springer, 2015, pp. 168–180.

[89] W. Liu, S. Chawla, J. Bailey, C. Leckie, K. Ramamohanarao, An efficient adversarial learning strategy for constructing robust classification boundaries, in: Australasian Joint Conference on Artificial Intelligence, Springer, 2012, pp. 649–660.

[90] S. R. Bulò, B. Biggio, I. Pillai, M. Pelillo, F. Roli, Randomized prediction games for adversarial machine learning, IEEE transactions on neural networks and learning systems 28 (2017) 2466–2478.

[91] L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, F. Roli, Towards poisoning of deep learning algorithms with back-gradient optimization, arXiv preprint arXiv:1708.08689 (2017).

[92] T. Alpcan, B. I. Rubinstein, C. Leckie, Large-scale strategic games and adversarial machine learning, in: Decision and Control (CDC), 2016 IEEE 55th Conference on, IEEE, 2016, pp. 4420–4426.

[93] L. Munoz-Gonzalez, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, F. Roli, Towards poisoning of deep learning algorithms with back-gradient optimization, in: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, ACM, 2017, pp. 27–38.

[94] J. Li, A. Madry, J. Peebles, L. Schmidt, Towards understanding the dynamics of generative adversarial networks, arXiv preprint arXiv:1706.09884 (2017).

[95] G. Louppe, K. Cranmer, Adversarial variational optimization of non-differentiable simulators, arXiv preprint arXiv:1707.07113 (2017).

[96] A. N. Bhagoji, D. Cullina, C. Sitawarin, P. Mittal, Enhancing robustness of machine learning systems via data transformations, in: Information Sciences and Systems (CISS), 2018 52nd Annual Conference on, IEEE, 2018, pp. 1–5.

[97] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, N. Usunier, Parseval networks: Improving robustness to adversarial examples, in: International Conference on Machine Learning, 2017, pp. 854–863.