Attacks on Machine Learning: Lurking Danger for Accountability

Katja Auernhammer, Ramin Tavakoli Kolagari, Markus Zoppelt
Nuremberg Institute of Technology, Faculty of Computer Science
Hohfederstrasse 40
Nuremberg, 90489, Germany
{katja.auernhammer, ramin.tavakolikolagari, markus.zoppelt}@th-nuernberg.de

Abstract

It is well-known that there is no safety without security. That being said, a sound investigation of security breaches on Machine Learning (ML) is a prerequisite for any safety concerns. Since attacks on ML systems and their impact on the security goals threaten the safety of an ML system, we discuss the impact attacks have on the ML models' security goals, which are rarely considered in published scientific papers.

The contribution of this paper is a non-exhaustive list of published attacks on ML models and a categorization of attacks according to their phase (training, after-training) and their impact on security goals. Based on our categorization we show that not all security goals have yet been considered in the literature, either because they were ignored or there are no publications on attacks targeting those goals specifically, and that some are difficult to assess, such as accountability. This is probably due to some ML models being a black box.

Introduction

During the last few years scientists and researchers have published a variety of different attacks on Machine Learning (ML) systems. However, the papers only rarely mention security goals—such as integrity, availability, confidentiality, reliability, authenticity, and accountability—that are endangered by these attacks. Even if a paper explicitly mentions the violation of a security goal it is not clear if the breach refers to the whole system in which the ML model is embedded or rather the ML model itself or parts of it.

The contribution of this paper is a non-exhaustive list of published attacks on ML and a derivation of different groups of attacks. We further elaborate on the breaches of known security goals (integrity, availability, confidentiality, etc.) caused by the listed attacks to justify our categorization and show the security goals mentioned in published papers about attacks on ML. Our categorization clarifies that there are some security goals, such as accountability, which are yet difficult to evaluate due to the complex operations within ML models.

Security Goals

The six main security goals as described in [21] are summarized as follows:

- **Confidentiality** ensures that private or confidential information is not made available or disclosed to unauthorized users, and that users can control (or influence) what information related to them may be collected, used, and to whom it is disclosed. Confidentiality is often implemented through cryptography / encryption.
- **Integrity** ensures that information is not changed (modified) or destroyed unauthorizedly. Integrity can be compromised even if the information or system produces the correct output.
- **Availability** ensures that a system works promptly, service is not denied to authorized users, and access to and use of information is timely and reliable.
- **Authenticity** is the characteristic of being genuine and verifiable and trustworthy. Authenticity is ensured through authentication processes that verify whether users are who they say they are (entity authenticity). Authenticity is often enabled through cryptography / cryptographic signatures.
- **Reliability** is the property of a system such that reliance can be justifiably placed on the service it delivers, i.e., the system adheres to the specification it was engineered to address.
- **Accountability** refers to the requirements for actions of an entity to be traced uniquely to that entity (e.g., non-repudiation of a communication that took place). Accountability allows a certain degree of transparency to what happened when and what was performed by whom.

Attacks on Machine Learning Algorithms

Important criteria that influence the applicability of certain attacks on ML models at this level of detail are the learning type (supervised, unsupervised, reinforcement learning) and if the algorithm undergoes lifelong learning. Different attacks are designed to target combinations of different criteria. The implications to the security goals of the ML model are equivalent to the security goals corresponding to the categorization of the attack.

In Table 1 the first column names the ML algorithm in alphabetical order, followed by the learning type and whether the model is capable of lifelong learning or not. Lifelong learning is a criterion that is often ignored by researchers

Table 1: Published attacks on ML categorized by ML algorithms. The listed ML algorithms are derived from the publications of the attacks, therefore, there might be attacks aimed at, e.g., neural networks in general but also attacks on specific sub-types of neural networks, e.g., convolutional neural networks. The columns “Learning Type” and “Lifelong Learning” do not solely refer to what the algorithm is capable of but to the premises the ML algorithm must meet to render the attack effective

ML Algorithm	Learning Type	Lifelong L.	Attack
Complete-linkage Hierarchical Clustering	Unsupervised	No	Poisoning Attack [9]
Single-Linkage Hierarchical Clustering	Unsupervised	No	Poisoning Attack [13]
			Obfuscation Attack [13, 14]
Decision Tree/Random Forest	Supervised	Yes/No	Poisoning Attack [46]
		No	Path-finding Attack [72]
			Model Inversion [26]
			Ateniese et al. Attack [4]
			Adversarial Examples [31, 52, 66]
Hidden Markov Model	Supervised	No	Ateniese et al. Attack [4]
k-Nearest Neighbors	Supervised	Yes/No	Poisoning Attack [46]
		No	Adversarial Examples [31]
k-Means Clustering	Unsupervised	No	Ateniese et al. Attack [4]
Linear Regression	Supervised	Yes/No	Poisoning Attack [8, 35, 41]
		No	Model Inversion [27]
			Lowd-Meek Attack [44, 72]
Logistic Regression	Supervised	No	Equation-solving Attack [49]
			Hyperparameter Stealing [73]
			Adversarial Examples [52, 70, 71]
Multi-class Logistic Regression	Supervised	No	Equation-solving Attack [49]
Maximum Entropy Models	Supervised	No	Lowd-Meek Attack [44]
Naive Bayes	Supervised	No	Classifier Evasion [3, 22]
			Lowd-Meek Attack [44]
Neural Network	Reinforcement Learning	Unclear	Strategically-timed Attack [40]
			Enchanting Attack [40]
			Adversarial Examples [33, 40]
Neural Network	Supervised	No	Model Inversion [26]
			Membership Inference [63]
			Hyperparameter Stealing Attack [73]
			Ateniese et al. Attack [4]
			Adversarial Examples [29, 31, 45, 52, 62, 70]
			Trojan Trigger [43]
Multi-layer Perceptron	Supervised	Yes/No	Poisoning Attack [46]
		No	Equation-solving Attack [49]
			Ateniese et al. Attack [4]
Convolutional Neural Network	Supervised	No	Side-channel Attack [74]
			Training Data Extraction [18]
			Adversarial Examples [50, 52, 70]
Recurrent Neural Network	Supervised	No	Training Data Extraction [18]
			Classifier Evasion [3]
			Adversarial Examples [57]
Support Vector Machine	Supervised	Yes/No	Poisoning Attack [12, 46]
			Adversarial Label Flips [76, 77]
		No	Hyperparameter Stealing [73]
			Lowd-Meek Attack [44, 72]
			Ateniese et al. Attack [4]
			Evasion Attack [3, 24, 30, 61, 66]
			Feature Deletion [28]
			Adversarial Examples [31, 52, 66, 71]

or at least not explicitly mentioned in papers. We complemented this information wherever necessary according to the definition in common text books. There are four possible values for lifelong learning: Yes, No, Yes/No (when both can be the case) and unclear (when we simply do not know). In the last column we list the attacks with corresponding literature.

We also identified attacks that are employable against several ML algorithms. Attacks we consider applicable to systems regardless of the ML algorithm, learning type, and lifelong learning capability are, for example, poisoning attacks [8, 46] as these attacks do not focus on the model but the training data; therefore, poisoning attacks are considered independent of the ML algorithm.

Another group of attacks that tamper with data fed into the ML model, and thus are applicable on a wide range of different ML algorithms, are adversarial examples [5, 6, 17, 34, 51, 59], evasion attacks [23, 78], and feature deletion attacks. These attacks exploit weaknesses in the ML model without changing the model itself by simply perturbing the input to falsify the output.

Shokri et al. [63] claim their attack, membership inference, to be generic, although they only apply it to classification algorithms. We also think the attack is only applicable to ML algorithms that are not capable of lifelong learning, as membership inference relies on computing multiple inputs via the ML model to extract information about the training data. If the model adapts with every given input, this approach can be aggravated.

Categorization of ML Attacks with Regard to Security Goals

In software security it is well-established to distinguish between attacks with regard to their effects on security goals (see Section "Security Goals"). The attacks described in Table 2 affect one or more security goals of a system (here: an ML component). A categorization of the published attacks according to security goals compiles an overview of clusters of similar attack scenarios as well as of missing but expected attack clusters. These gaps in the categorization of attacks may result from unknown publications about attacks on ML components, from unpublished attacks or attacks that have not yet been executed but which are all conceivable and therefore executable in principle. Therefore, these gaps in the categorization are particularly revealing.

Of particular relevance for the categorization of attacks developed here is the violation of security goals, which affect the ML component as a whole. Thus, the violation of the integrity for an ML component means that the ML component itself is changed (in some form). In the publications on the attacks on ML components analyzed here (and also listed in Table 2), statements are partly made on the violations of the security goals, but these sometimes refer (only) to partial areas of an attack. Thus, the attack adversarial examples [69], which manipulates data fed into the model, targets—according to the authors—integrity, namely the integrity of the input data; as the integrity of the ML model itself is not attacked because it has not been changed, it is not categorized in Table 2 under integrity.

Table 2 shows our mapping of the analyzed attacks listed in Table 1 to the six security goals described in the Security Goals section. While Table 1 focused on the ML algorithms Table 2 brings the attacks into focus. The assignment in Table 2 is based on the description of the attacks in the respective publications. In the table, an "X" indicates which security goal (related to the ML component as a whole) is affected by which attack.

In addition, many attacks have been published that relate to pre- and post-processing units of ML components (their environment). These attacks do not differ from those on traditional software, therefore they are not described in this paper.

An obvious peculiarity of ML components compared to traditional software is their training, so there are two essential phases in their life cycle: the training phase (T) and the deployment phase that we prefer to call the after-training phase (A), as this also considers lifelong learning ML algorithms, which are trained with every input even after deployment. This continuous learning process makes attacks in deployment time possible, which are also applicable in training time (such as poisoning attacks [60]) and, on the other hand, disables the applicability of attacks that require a fixed target model (e.g., model inversion [26]).

Unlike previous research (e.g., [7, 55]) we do not consider whether an attack is targeted, whether the opponent causes a certain wrong output or not, whether a wrong output is generated, or whether the opponent has white box or black box knowledge. At this point we also do not distinguish between different types of learning (supervised, unsupervised, reinforcement learning). Considering all these kinds of criterion, a blurred categorization would be created that contradicts a clear distinction between attacks. Instead, we propose considering the above criteria within each of our main groups in order to add further dimensions and form sub-groups. This is not within the scope of this paper, although we consider the learning type in Table 1, which can be used as a starting point for further investigations.

By analyzing the security goals that are breached by the attacks and the time the attack takes place, we can create different categories of attacks. The names of the categories are derived from whether the attack takes place during training time (T) or after-training time (A) followed by a dash (-) and the first one or two letters of the main security goals, which are breached by the attacks. Grey "X"s indicate the main assignments of attacks to security goals.

First of all, it is noticeable that all attacks at training time affect both integrity and reliability. This also makes sense immediately: if only the integrity was corrupted during training time, the system could be corrected conform to the specification via the existing reliability. If only the reliability was corrupted, the unchanged behavior would result in a difference to the specification, which would result in a correction of the specification. Only a simultaneous attack on both security goals can therefore be successful during the training phase. Confidentiality is not a main security goal for attacks during the training phase, but most of the identified attacks have attacked the confidentiality as well. However, success-

Table 2: Mapping of published attacks on ML on the security goals violated. The attacks are categorized according to the security goals they breach. The first column "Att. Cat." (Attack Category) labels the categories. The names are derived from the time of the ML algorithm lifecycle (Training, After-training) the attacks take place and the security goals the attack brakes that are most relevant for the specified category

Att. Cat.	**Published Attacks**	**Confiden-tiality**	**Availa-bility**	**Integrity**	**Reliability**	**Authen-ticity**	**Accoun-tability**
T-IR	Poisoning Attack [60]	X [14, 47]	[10, 13, 14, 35, 39, 47]	X [10, 14, 35, 36, 39, 47, 55, 67]	X		
	Adversarial Label Flips [76]	X		X [56]	X		
	Strategically-timed Attack [40]	X		X	X		
	Enchanting Attack [40]	X		X	X		
	Obfuscation Attack [13]			X [13]	X		
A-IR	Trojan Trigger [43]	X		X	X		
A-C	Model Inversion [26]	X [26, 27, 32, 56, 72, 75]	X				
	Membership Inference [63]	X [63, 65]	X				
	Side-channel Attack [74]	X [74]	X				
	Lowd-Meek Attack [44]	X		[55]			
	Training Data Extraction [18]	X [18]	X				
	Ateniese et al. Attack [4]	X [4]	X				
	Path-finding Attack [49]	X	X				
	Equation-solving Attack [49]	X					
	Hyperparameter Stealing [73]	X [73]					
A-R	Classifier Evasion [11]		X [7, 48]	[20, 48, 55, 68]	X		
	Adversarial Examples [69]		X [15]	[15, 17, 52, 53, 54, 55]	X [25, 64]		
	Feature Deletion [28]		X		X		

ful attacks during the training phase that relate exclusively to integrity and reliability would also be conceivable. Attacking the security goal availability makes no sense during the training phase.

Attacks on integrity and reliability during the deployment phase are theoretically meaningful and have been published pertinently. They represent the mirroring of attacks on integrity and reliability from the training phase. An essential group with a particularly large number of published attacks in the deployment phase refers to confidentiality. The fact that these attacks are often accompanied by restrictions in availability is rather a side effect than a main aspect. A category of attacks on ML components that mainly refers to availability (think of DoS attacks on traditional software) makes little sense in theory and has not been published. The frequently cited adversarial examples attack group is among others in the category of reliability attacks during the deployment phase; typically, integrity is not corrupted because the ML components themselves are not modified.

The lack of assignments to the security goals authenticity and accountability are also particularly informative. In our research we could not find any attacks on these security goals of the ML components. Authenticity is usually implemented in the environmental components surrounding an ML component. This will probably change in the future, however, when comprehensive tasks will be implemented in a network of ML components and it becomes necessary to establish the ML components as mission-critical communication partners. Accountability of ML is considered—even in the community of ML experts—to be mostly inaccessible (especially with the so-called black box ML components such as deep neural networks), because these components cannot be read like traditional software and cannot be semantically deduced from the structure. Nevertheless, we believe that a new field of attacks on ML components will open up in this field in the future because initiatives such as eXplainable AI (layer-wise relevance propagation [16], Black Box Explanations through Transparent Approximations (BETA) [37], LIME [58], Generalized Additive Model (GAM) [19], etc.) and the political demand for comprehensible AI decisions will ensure greater comprehensibility in the area of the black box ML, which will ultimately also help the attackers.

The Peculiarity of Accountability

It is yet unclear, how the concept of accountability applies to ML. Accountability in traditional software engineering means an action can always be retraced to the entity performing the action. An entity is usually a human or a digital agent, however, the definition of an entity is not clear in the field of ML. An entity could be an input feature which leads to a certain output of the ML model (this meets the definition made by Papernot et al. [56]). An entity could also be an element within in the ML model, e.g., each single neuron within a neural network, which makes its own decision that influences the final output of the model. From a different point of view even the software developer could be considered the entity.

The entity, which can not deny an action, is ultimately relevant in a legal context, namely in case of finding the party liable for a specific action. It is not relevant, however, how a single element of an algorithm contributed to the system's decision, but whether the wrong decision was caused due to faulty training, biases in the training data or malicious attacks.

We find that there is no clear definition of accountability and that it is difficult to transfer existing definitions to the field of ML. In order to guarantee accountability at all, changes in the system, e.g., in traditional software this could be changes in the database, must be recorded. Without a form of audit that promises some form of tracing, accountability cannot be broken, because the goal was not even reached in the first place. With a ML system, the changes within a system do not necessarily have to be recorded. Rather the decisions of the system or of parts of the system should be made assignable to a distinct entity.

In the context of ML, a distinction between accountability and liability should be considered. Both focus on retracing an action to an entity. Liability, however, concentrates on the assignment of blame or debt relief of individual entities and is also possible without an audit of the actions and decision made by inner components within the ML algorithm. For liability it is sufficient to record the final decision of the ML system solely.

Accountability, on the other hand, is only possible by logging the internal processes. The definition of an "entity", however, is still unclear. Furthermore, logging requires a certain understanding of the model, which is difficult up until now. However, if ML algorithms become comprehensible in the future, accountability could be achievable and this also means that accountability—as a security goal—can be broken by attackers.

Assume it will be possible to identify which nodes in a neural network are responsible for a particular decision. E.g., we know which nodes in an image recognition system are responsible for detecting certain objects, such as stop signs. If these nodes are regarded as entities, they can be made accountable for their decisions. Accountability allows ML algorithms to be developed and validated more efficiently maybe even to the point where they become similar to the code of traditional software development. This is desirable in any case, as it greatly simplifies development and troubleshooting. If this knowledge about accountability is leaked, adversaries can also take advantage of it and launch more targeted attacks, which might ultimately also target accountability. A breach in accountability will most likely be the first step to sophisticated attacks that violate other security goals as well.

It is unclear what types of attacks might be possible once ML models can be fully explained to humans, though.

Related Work

Barreno et al. [7] give relevant properties they consider important when conducting attacks on ML. The properties are grouped into three categories: the influence of the attack on the target system, the specificity (targeted or untargeted) and the security violation (integrity, availability). Their paper focuses mostly on countermeasures against attacks. Papernot et al. [55] also review attacks and distinguish them into black box and white box attacks. They focus on attacks on classification algorithms and list theoretical countermeasures. Liu et al. [42] also discuss different attacks and propose interesting points to consider in future research. Biggio et al. [15] take a different view on attacks on ML. They focus on how the field has developed during the years since its first mention in 2004. They also review published countermeasures.

Alabdulmohsin et al. [2] sort attacks into causative or exploratory attacks. A survey of attacks against deep learning in computer vision was conducted by Akhtar and Mian [1]. They list several published countermeasures against adversarial examples. Laskov and Kloft [38] propose a "framework for quantitative security analysis of ML models".

Conclusion

In this paper we give an overview of the current state-of-the-art ML algorithms and their respective attacks. This list is especially interesting when considering some of the more critical fields ML is used in, such as autonomous driving. Autonomous driving uses ML models in safety-critical applications. Ignoring known attacks on pertinent ML algorithms is hazardous as human life is at stake. Likewise, regular software development, security by design has to be applied to the development of ML algorithms as well.

We also propose a classification of published attacks on ML models based on security goals and life cycle phase.

Our research shows that accountability is not covered by literature as there have not yet been any attacks published. This is probably due to the fact that accountability for ML is difficult to attack as ML models are yet beyond human understanding and, therefore, the security goal is not compulsory.

Although, there are already some papers working on a solution to improve comprehensibility of ML models, we think there is still a long way to go until humans are able to completely understand ML models. If accountability can be guaranteed for all kinds of ML models this will enable a wide range of new yet unknown attacks.

Further research will elaborate the implications of vulnerable ML models. It will also discuss whether and how the security goal accountability can be transferred to the field of ML and if proper accountability of ML models has to be considered in liability claims.

Acknowledgement

Katja Auernhammer and Markus Zoppelt were supported by the BayWISS Consortium Digitization.

References

[1] Naveed Akhtar and Ajmal Mian. "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey". In: *IEEE Access* 6 (Jan. 2018), pp. 14410–14430. ISSN: 21693536. DOI: 10.1109/ACCESS.2018.2807385. arXiv: 1801.00553.

[2] Ibrahim M. Alabdulmohsin, Xin Gao, and Xiangliang Zhang. "Adding Robustness to Support Vector Machines Against Adversarial Reverse Engineering". In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management - CIKM '14*. New York, New York, USA: ACM Press, 2014, pp. 231–240. ISBN: 9781450325981. DOI: 10.1145/2661829.2662047.

[3] Mark Anderson, Andrew Bartolo, and Pulkit Tandon. "Crafting Adversarial Attacks on Recurrent Neural Networks". In: (2017).

[4] Giuseppe Ateniese, Giovanni Felici, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, and Domenico Vitali. "Hacking Smart Machines with Smarter Ones: How to Extract Meaningful Data from Machine Learning Classifiers". In: (2013). ISSN: 1747-8405. DOI: 10.1504/IJSN.2015.071829. arXiv: 1306.4447.

[5] Shumeet Baluja and Ian Fischer. "Adversarial Transformation Networks: Learning to Generate Adversarial Examples". In: (2017). arXiv: 1703.09387.

[6] Shumeet Baluja and Ian Fischer. "Learning to Attack: Adversarial Transformation Networks". In: *Association for the Advancement of Artificial Intelligence - AAAI'18*. 2018.

[7] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D. Joseph, and J. D. Tygar. "Can machine learning be secure?" In: *Proceedings of the 2006 ACM Symposium on Information, computer and communications security - ASIACCS '06*. New York, USA: ACM Press, 2006. ISBN: 1595932720. DOI: 10.1145/1128817.1128824.

[8] Alex Beatson, Zhaoran Wang, and Han Liu. "Blind Attacks on Machine Learners". In: *30th Conference on Neural Information Processing Systems (NIPS 2016)* (2016), pp. 2397–2405. ISSN: 10495258.

[9] Battista Biggio, Samuel Rota Bulò, Ignazio Pillai, Michele Mura, Eyasu Zemene Mequanint, Marcello Pelillo, and Fabio Roli. "Poisoning Complete-Linkage Hierarchical Clustering". In: ed. by Ana Fred, Terry M. Caelli, Robert P. W. Duin, Aurélio C. Campilho, and Dick de Ridder. Vol. 3138. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, Aug. 2004. ISBN: 978-3-540-22570-6. DOI: 10.1007/b98738. arXiv: 9780201398298.

[10] Battista Biggio, Igino Corona, Giorgio Fumera, Giorgio Giacinto, and Fabio Roli. "Bagging classifiers for fighting poisoning attacks in adversarial classification tasks". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 6713 LNCS (2011), pp. 350–359. ISSN: 03029743. DOI: 10.1007/978-3-642-21557-5_37.

[11] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. "Evasion Attacks against Machine Learning at Test Time". In: *ECML PKDD* (2013), pp. 387–402. DOI: 10.1007/978-3-642-40994-3_25.

[12] Battista Biggio, Blaine Nelson, and Pavel Laskov. "Poisoning Attacks against Support Vector Machines". In: *Proceedings of the 29 th International Conference on Machine Learning* (June 2012). arXiv: 1206.6389.

[13] Battista Biggio, Ignazio Pillai, Samuel Rota Bulò, Davide Ariu, Marcello Pelillo, and Fabio Roli. "Is data clustering in adversarial settings secure?" In: *Proceedings of the*

2013 ACM workshop on Artificial intelligence and security - AISec '13 (2013), pp. 87–98. ISSN: 15437221. DOI: 10.1145/2517312.2517321.

[14] Battista Biggio, Konrad Rieck, Davide Ariu, Christian Wressnegger, Igino Corona, Giorgio Giacinto, and Fabio Roli. "Poisoning behavioral malware clustering". In: *Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop - AISec '14*. New York, USA: ACM Press, Nov. 2014, pp. 27–36. ISBN: 9781450331531. DOI: 10.1145/2666652.2666666.

[15] Battista Biggio and Fabio Roli. "Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning". In: *Pattern Recognition* 84 (Dec. 2017), pp. 317–331. ISSN: 00313203. DOI: 10.1016/j.patcog.2018.07.023. arXiv: 1712.03141.

[16] Alexander Binder, Sebastian Bach, Gregoire Montavon, Klaus Robert Müller, and Wojciech Samek. "Layer-wise relevance propagation for deep neural network architectures". In: *Lecture Notes in Electrical Engineering* 376 (2016), pp. 913–922. ISSN: 18761119. DOI: 10.1007/978-981-10-0557-2_87.

[17] Wieland Brendel, Jonas Rauber, and Matthias Bethge. "Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models". In: (Dec. 2017). arXiv: 1712.04248.

[18] Nicholas Carlini, Chang Liu, Jernej Kos, Úlfar Erlingsson, and Dawn Song. "The Secret Sharer: Measuring Unintended Neural Network Memorization & Extracting Secrets". In: (Feb. 2018). arXiv: 1802.08232.

[19] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. "Intelligible Models for HealthCare". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15* (2015), pp. 1721–1730. ISSN: 1869-0327. DOI: 10.1145/2783258.2788613.

[20] Lingwei Chen, Yanfang Ye, and Thirimachos Bourlai. "Adversarial machine learning in malware detection: Arms race between evasion attack and defense". In: *Proceedings - 2017 European Intelligence and Security Informatics Conference, EISIC 2017* 2017-Janua (2017), pp. 99–106. DOI: 10.1109/EISIC.2017.21.

[21] Fabiano Dalpiaz, Elda Paja, and Paolo Giorgini. *Security Requirements Engineering: Designing Secure Socio-Technical Systems*. MIT Press, 2016, p. 224. ISBN: 0262034212.

[22] Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. "Adversarial classification". In: *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*. New York, USA: ACM Press, 2004. DOI: 10.1145/1014052.1014066.

[23] Hung Dang, Yue Huang, and Ee-Chien Chang. "Evading Classifiers by Morphing in the Dark". In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security - CCS '17*. New York, USA: ACM Press, 2017, pp. 119–133. ISBN: 9781450349468. DOI: 10.1145/3133956.3133978. arXiv: 1705.07535.

[24] Ambra Demontis, Paolo Russu, Battista Biggio, Giorgio Fumera, and Fabio Roli. "On security and sparsity of linear classifiers for adversarial settings". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10029 LNCS (2016), pp. 322–332. ISSN: 16113349. DOI: 10.1007/978-3-319-49055-7_29. arXiv: 1709.00045.

[25] Alhussein Fawzi, Seyed Mohsen Moosavi-Dezfooli, and Pascal Frossard. "The Robustness of Deep Networks: A Geometrical Perspective". In: *IEEE Signal Processing Magazine* 34.6 (2017), pp. 50–62. ISSN: 10535888. DOI: 10.1109/MSP.2017.2740965.

[26] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. "Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS '15*. New York, USA: ACM Press, 2015, pp. 1322–1333. ISBN: 9781450338325. DOI: 10.1145/2810103.2813677.

[27] Matt Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. "Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing". In: *Proceedings of the 23rd USENIX Security Symposium* (2014), pp. 17–32.

[28] Amir Globerson and Sam Roweis. "Nightmare at test time: robust learning by feature deletion". In: *Proceedings of the 23rd international conference on Machine learning* (2006), pp. 353–360. DOI: 10.1145/1143844.1143889.

[29] Abigail Graese, Andras Rozsa, and Terrance E. Boult. "Assessing threat of adversarial examples on deep neural networks". In: *Proceedings - 2016 15th IEEE International Conference on Machine Learning and Applications, ICMLA 2016* (2017), pp. 69–74. DOI: 10.1109/ICMLA.2016.44. arXiv: 1610.04256.

[30] Yi Han and Benjamin I. P. Rubinstein. "Adequacy of the Gradient-Descent Method for Classifier Evasion Attacks". In: (2017). arXiv: 1704.01704.

[31] Jamie Hayes and George Danezis. "Machine Learning as an Adversarial Service: Learning Black-Box Adversarial Examples". In: (2017). arXiv: 1708.05207.

[32] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. "Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning". In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security - CCS '17*. New York, USA: ACM Press, 2017, pp. 603–618. ISBN: 9781450349468. DOI: 10.1145/3133956.3134012. arXiv: 1702.07464.

[33] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. "Adversarial Attacks on Neural Network Policies". In: (Feb. 2017). arXiv: 1702.02284.

[34] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. "Query-Efficient Black-box Adversarial Examples". In: (Dec. 2017). arXiv: 1712.07113.

[35] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. "Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning". In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2018, pp. 19–35. ISBN: 978-1-5386-4353-2. DOI: 10.1109/SP.2018.00057. arXiv: 1804.00308.

[36] Ricky Laishram and Vir Virander Phoha. "Curie: A method for protecting SVM Classifier from Poisoning Attack". In: (June 2016). arXiv: 1606.01584.

[37] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. "Interpretable & Explorable Approximations of Black Box Models". In: (2017). arXiv: 1707.01154.

[38] Pavel Laskov and Marius Kloft. "A framework for quantitative security analysis of machine learning". In: *Proceedings of the 2nd ACM workshop on Security and artificial intelligence - AISec '09*. New York, New York, USA: ACM Press, 2009. ISBN: 9781605587813. DOI: 10.1145/1654988.1654990.

[39] Bo Li, Yining Wang, Aarti Singh, and Yevgeniy Vorobeychik. "Data Poisoning Attacks on Factorization-Based Collaborative Filtering". In: *29th Conference on Neural Information Processing Systems (NIPS 2016)* Nips (Aug. 2016). ISSN: 10495258. arXiv: 1608.08182.

[40] Yen Chen Lin, Zhang Wei Hong, Yuan Hong Liao, Meng Li Shih, Ming Yu Liu, and Min Sun. "Tactics of adversarial attack on deep reinforcement learning agents". In: *IJCAI International Joint Conference on Artificial Intelligence* (2017), pp. 3756–3762. ISSN: 10450823. arXiv: 1703.06748.

[41] Chang Liu, Bo Li, Yevgeniy Vorobeychik, and Alina Oprea. "Robust Linear Regression Against Training Data Poisoning". In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security - AISec '17* (2017), pp. 91–102. DOI: 10.1145/3128572.3140447.

[42] Qiang Liu, Pan Li, Wentao Zhao, Wei Cai, Shui Yu, and Victor C.M. Leung. "A survey on security threats and defensive techniques of machine learning: A data driven view". In: *IEEE Access* 6 (2018), pp. 12103–12117. ISSN: 21693536. DOI: 10.1109/ACCESS.2018.2805680.

[43] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Authors Yingqi Liu, Weihang Wang, and Xiangyu Zhang. "Trojaning Attack on Neural Networks". In: *NDSS 2018 (Network and Distributed System Security Symposium)* (Feb. 2018). DOI: 10.14722/ndss.2018.23291.

[44] Daniel Lowd and Christopher Meek. "Adversarial learning". In: *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05* (2005). DOI: 10.1145/1081870.1081950.

[45] Seyed Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. "Universal adversarial perturbations". In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-January (2017), pp. 86–94. ISSN: 1063-6919. DOI: 10.1109/CVPR.2017.17. arXiv: 1705.09554.

[46] Mehran Mozaffari-Kermani, Susmita Sur-Kolay, Anand Raghunathan, and Niraj K. Jha. "Systematic poisoning attacks on and defenses for machine learning in healthcare". In: *IEEE Journal of Biomedical and Health Informatics* 19.6 (2015), pp. 1893–1905. ISSN: 21682194. DOI: 10.1109/JBHI.2014.2344095.

[47] Luis Muñoz-González, Battista Biggio, Ambra Demontis, Andrea Paudice, Vasin Wongrassamee, Emil C. Lupu, and Fabio Roli. "Towards Poisoning of Deep Learning Algorithms with Back-gradient Optimization". In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security - AISec '17*. New York, SA: ACM Press, 2017, pp. 27–38. ISBN: 9781450352024. DOI: 10.1145/3128572.3140451. arXiv: 1708.08689.

[48] Blaine Nelson, Marco Barreno, Fuching Jack Chi, Anthony D. Joseph, Benjamin I.P. Rubinstein, Udam Saini, Charles Sutton, J. D. Tygar, and Kai Xia. "Exploiting machine learning to subvert your spam filter". In: *In Proceedings of the First Workshop on Large-scale Exploits and Emerging Threats (LEET)* April (2008), Article 7.

[49] Tam N. Nguyen. "Attacking Machine Learning models as part of a cyber kill chain". In: (2017). arXiv: 1705.00564.

[50] Andrew P. Norton and Yanjun Qi. "Adversarial-Playground: A visualization suite showing how adversarial examples fool deep learning". In: *2017 IEEE Symposium on Visualization for Cyber Security (VizSec)*. Vol. 2017-Octob. IEEE, Oct. 2017. ISBN: 978-1-5386-2693-1. DOI: 10.1109/VIZSEC.2017.8062202. arXiv: 1708.00807.

[51] Nicolas Papernot. "Characterizing the Limits and Defenses of Machine Learning in Adversarial Settings". In: (2018).

[52] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. "Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples". In: (2016). arXiv: 1605.07277.

[53] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. "Practical Black-Box Attacks against Machine Learning". In: (Feb. 2016). DOI: 10.1145/3052973.3053009. arXiv: 1602.02697.

[54] Nicolas Papernot, Patrick Mcdaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. "The limitations of deep learning in adversarial settings". In: *Proceedings - 2016 IEEE European Symposium on Security and Privacy, EURO S and P 2016* (2016), pp. 372–387. DOI: 10.1109/EuroSP.2016.36. arXiv: 1511.07528.

[55] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. "SoK: Towards the Science of Security and Privacy in Machine Learning". In: (Nov. 2016). arXiv: 1611.03814.

[56] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael P. Wellman. "SoK: Security and Privacy in Machine Learning". In: *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, Apr. 2018, pp. 399–414. ISBN: 978-1-5386-4228-3. DOI: 10.1109/EuroSP.2018.00035. arXiv: 1611.03814.

[57] Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. "Crafting adversarial input sequences for recurrent neural networks". In: *MILCOM 2016 - 2016 IEEE Military Communications Conference*. IEEE, Nov. 2016, pp. 49–54. ISBN: 978-1-5090-3781-0. DOI: 10.1109/MILCOM.2016.7795300. arXiv: 1604.08275.

[58] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 2016), pp. 1135–1144. ISSN: 9781450321389. DOI: 10.1145/2939672.2939778. arXiv: 1602.04938.

[59] Amir Rosenfeld, Richard Zemel, and John K. Tsotsos. "The Elephant in the Room". In: (2018). arXiv: 1808.03305.

[60] Benjamin I.P. Rubinstein, Blaine Nelson, Ling Huang, Anthony D Joseph, Shing-hon Lau, Satish Rao, Nina Taft, and J. D. Tygar. "ANTIDOTE: Understanding and Defending against Poisoning of Anomaly Detectors". In: *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference - IMC '09*. New York, New York, USA: ACM Press, Nov. 2009. ISBN: 9781605587714. DOI: 10.1145/1644893.1644895.

[61] Paolo Russu, Ambra Demontis, Battista Biggio, Giorgio Fumera, and Fabio Roli. "Secure Kernel Machines against Evasion Attacks". In: *Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security - ALSec '16* (2016), pp. 59–69. DOI: 10.1145/2996758.2996771.

[62] Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. "Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks". In: (Apr. 2018). arXiv: 1804.00792.

[63] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. "Membership Inference Attacks Against Machine Learning Models". In: *Proceedings - IEEE Symposium on Security and Privacy* (2017), pp. 3–18. ISSN: 10816011. DOI: 10.1109/SP.2017.41. arXiv: 1610.05820.

[64] D.B. Skillicorn. "Adversarial Knowledge Discovery". In: *IEEE Intelligent Systems* 24.6 (2009), pp. 1–13. ISSN: 1541-1672. DOI: 10.1109/MIS.2009.108.

[65] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. "Machine Learning Models that Remember Too Much". In: (2017). ISSN: 15437221. DOI: 10.1145/3133956.3134077. arXiv: 1709.07886.

[66] Nedim Šrndić and Pavel Laskov. "Practical evasion of a learning-based classifier: A case study". In: *Proceedings - IEEE Symposium on Security and Privacy* (2014), pp. 197–211. ISSN: 10816011. DOI: 10.1109/SP.2014.20.

[67] Jacob Steinhardt, Pang Wei Koh, and Percy Liang. "Certified Defenses for Data Poisoning Attacks". In: (June 2017). ISSN: 10495258. arXiv: 1706.03691.

[68] Rock Stevens, Octavian Suciu, Andrew Ruef, Sanghyun Hong, Michael Hicks, and Tudor Dumitraş. "Summoning Demons: The Pursuit of Exploitable Bugs in Machine Learning". In: (Jan. 2017). arXiv: 1701.04739.

[69] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. "Intriguing properties of neural networks". In: (Dec. 2013), pp. 1–10. ISSN: 15499618. DOI: 10.1021/ct2009208. arXiv: 1312.6199.

[70] Pedro Tabacof and Eduardo Valle. "Exploring the space of adversarial images". In: *Proceedings of the International Joint Conference on Neural Networks* 2016-Octob.1 (2016), pp. 426–433. DOI: 10.1109/IJCNN.2016.7727230. arXiv: 1510.05328.

[71] Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. "The Space of Transferable Adversarial Examples". In: (2017). arXiv: 1704.03453.

[72] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. "Stealing Machine Learning Models via Prediction APIs". In: *Proceedings of the 25th USENIX Security Symposium* 94.3 (Sept. 2016), pp. 601–618. ISSN: 2469-9985. DOI: 10.1103/PhysRevC.94.034301. arXiv: 1609.02943.

[73] Binghui Wang and Neil Zhenqiang Gong. "Stealing Hyperparameters in Machine Learning". In: *Proceedings - IEEE Symposium on Security and Privacy* 2018-May.May (2018), pp. 36–52. ISSN: 10816011. DOI: 10.1109/SP.2018.00038. arXiv: 1802.05351.

[74] Lingxiao Wei, Yannan Liu, Bo Luo, Yu Li, and Qiang Xu. "I Know What You See: Power Side-Channel Attack on Convolutional Neural Network Accelerators". In: (2018). arXiv: 1803.05847.

[75] Xi Wu, Matthew Fredrikson, Somesh Jha, and Jeffrey F. Naughton. "A methodology for formalizing model-inversion attacks". In: *Proceedings - IEEE Computer Security Foundations Symposium* 2016-Augus (2016). ISSN: 19401434. DOI: 10.1109/CSF.2016.32.

[76] Han Xiao, Huang Xiao, and Claudia Eckert. "Adversarial label flips attack on support vector machines". In: *Frontiers in Artificial Intelligence and Applications* 242.4 (2012), pp. 870–875. ISSN: 09226389. DOI: 10.3233/978-1-61499-098-7-870.

[77] Huang Xiao, Battista Biggio, Blaine Nelson, Han Xiao, Claudia Eckert, and Fabio Roli. "Support vector machines under adversarial label contamination". In: *Neurocomputing* 160 (2015), pp. 53–62. ISSN: 18728286. DOI: 10.1016/j.neucom.2014.08.081.

[78] Zhizhou Yin, Fei Wang, Wei Liu, and Sanjay Chawla. "Sparse Feature Attacks in Adversarial Learning". In: *IEEE Transactions on Knowledge and Data Engineering* 30.6 (2018), pp. 1164–1177. ISSN: 10414347. DOI: 10.1109/TKDE.2018.2790928.