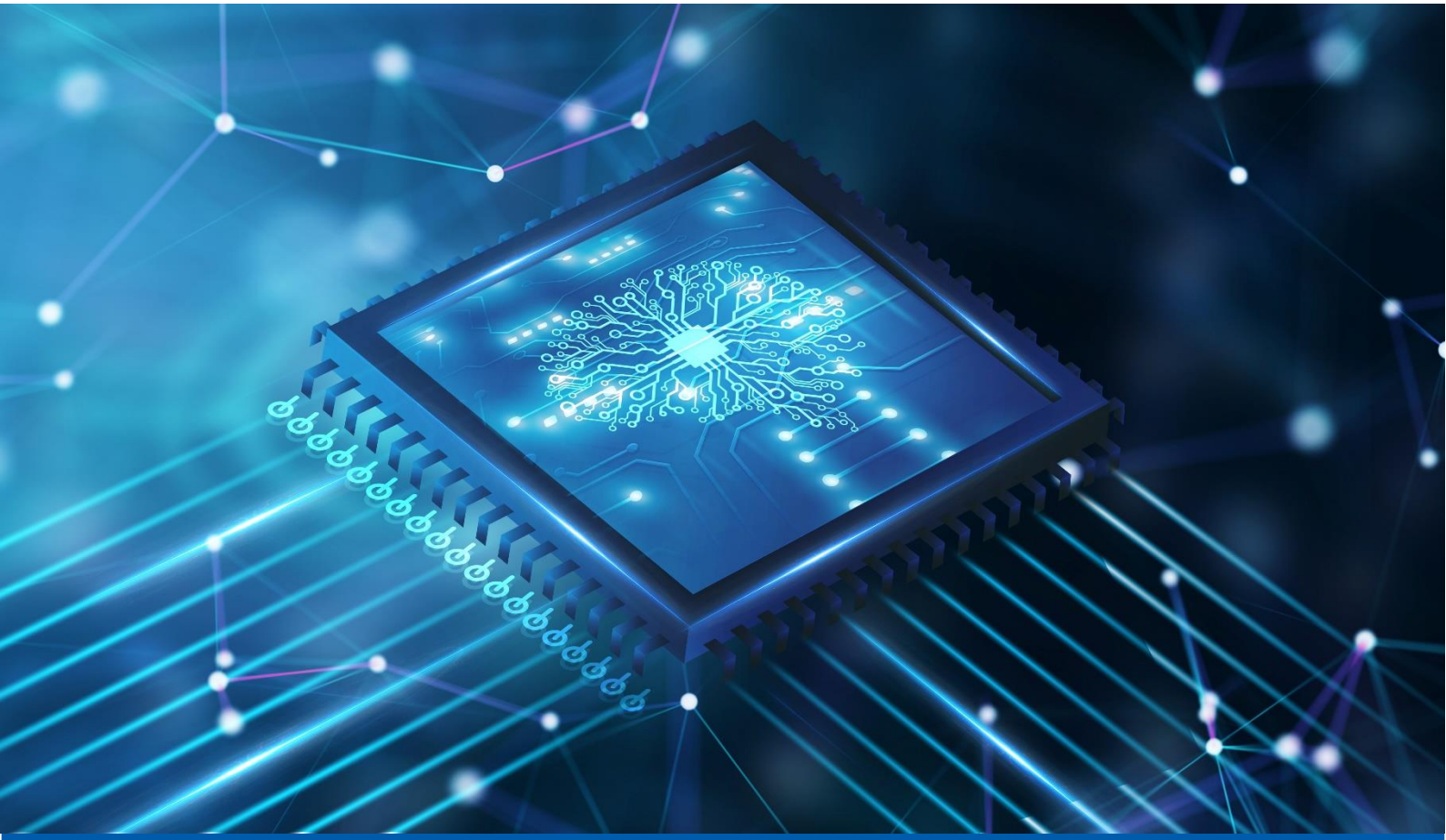
EUROPEAN UNION AGENCY
FOR CYBERSECURITY



SECURING MACHINE LEARNING ALGORITHMS

DECEMBER 2021

ABOUT ENISA

The European Union Agency for Cybersecurity, ENISA, is the Union's agency dedicated to achieving a high common level of cybersecurity across Europe. Established in 2004 and strengthened by the EU Cybersecurity Act, the European Union Agency for Cybersecurity contributes to EU cyber policy, enhances the trustworthiness of ICT products, services and processes with cybersecurity certification schemes, cooperates with Member States and EU bodies, and helps Europe prepare for the cyber challenges of tomorrow. Through knowledge sharing, capacity building and awareness raising, the Agency works together with its key stakeholders to strengthen trust in the connected economy, to boost resilience of the Union's infrastructure, and, ultimately, to keep Europe's society and citizens digitally secure. More information about ENISA and its work can be found here: www.enisa.europa.eu.

CONTACT

For contacting the authors please use info@enisa.europa.eu
For media enquiries about this paper, please use press@enisa.europa.eu

EDITORS

Apostolos Malatras, Ioannis Agrafiotis, Monika Adamczyk, ENISA

ACKNOWLEDGEMENTS

We would like to thank the Members and Observers of the ENISA ad hoc Working Group on Artificial Intelligence for their valuable input and feedback.

LEGAL NOTICE

Notice must be taken that this publication represents the views and interpretations of ENISA, unless stated otherwise. This publication should not be construed to be a legal action of ENISA or the ENISA bodies unless adopted pursuant to the Regulation (EU) No 2019/881.
This publication does not necessarily represent state-of the-art and ENISA may update it from time to time.

Third-party sources are quoted as appropriate. ENISA is not responsible for the content of the external sources including external websites referenced in this publication.

This publication is intended for information purposes only. It must be accessible free of charge. Neither ENISA nor any person acting on its behalf is responsible for the use that might be made of the information contained in this publication.

COPYRIGHT NOTICE

© European Union Agency for Cybersecurity (ENISA), 2021
Reproduction is authorised provided the source is acknowledged.

Copyright for the image on the cover: © Shutterstock
For any use or reproduction of photos or other material that is not under the ENISA copyright, permission must be sought directly from the copyright holders.

ISBN: 978-92-9204-543-2 – DOI: 10.2824/874249 - Catalogue Nr.: TP-06-21-153-EN-N

TABLE OF CONTENTS

EXECUTIVE SUMMARY 3

1. INTRODUCTION 4

1.1 OBJECTIVES 4

1.2 METHODOLOGY 4

1.3 TARGET AUDIENCE 5

1.4 STRUCTURE 6

2. MACHINE LEARNING ALGORITHMS TAXONOMY 7

2.1 MAIN DOMAIN AND DATA TYPES 8

2.2 LEARNING PARADIGMS 9

2.3 NAVIGATING THE TAXONOMY 10

2.4 EXPLAINABILITY AND ACCURACY 10

2.5 AN OVERVIEW OF AN END-TO-END MACHINE LEARNING LIFECYCLE 11

3. ML THREATS AND VULNERABILITIES 13

3.1 IDENTIFICATION OF THREATS 13

3.2 VULNERABILITIES MAPPED TO THREATS 16

4. SECURITY CONTROLS 18

4.1 SECURITY CONTROLS RESULTS 18

5. CONCLUSION 26

A ANNEX: TAXONOMY OF ALGORITHMS 28

B ANNEX: MAPPING SECURITY CONTROLS TO THREATS 34

C ANNEX: IMPLEMENTING SECURITY CONTROLS 38

D ANNEX: REFERENCES 43

EXECUTIVE SUMMARY

The vast developments in digital technology influence every aspect of our daily lives. Emerging technologies, such as Artificial Intelligence (AI), which are in the epicentre of the digital evolution, have accelerated the digital transformation contributing in social and economic prosperity. However, the application of emerging technologies and AI in particular, entails perils that need to be addressed if we are to ensure a secure and trustworthy environment. In this report, we focus on the most essential element of an AI system, which are machine learning algorithms. We review related technological developments and security practices to identify emerging threats, highlight gaps in security controls and recommend pathways to enhance cybersecurity posture in machine learning systems.

Based on a systematic review of relevant literature on machine learning, we provide a taxonomy for machine learning algorithms, highlighting core functionalities and critical stages. The taxonomy sheds light on main data types used by algorithms, the type of training these algorithms entail (supervised, unsupervised) and how output is shared with users. Particular emphasis is given to the explainability and accuracy of these algorithms. Next, the report presents a detailed analysis of threats targeting machine learning systems. Identified threats include inter alia, data poisoning, adversarial attacks and data exfiltration. All threats are associated to particular functionalities of the taxonomy that they exploit, through detailed tables. Finally, we examine mainstream security controls described in widely adopted standards, such as ISO 27001 and NIST Cybersecurity framework, to understand how these controls can effectively detect, deter and mitigate harms from the identified threats. To perform our analysis, we map all the controls to the core functionalities of machine learning systems that they protect and to the vulnerabilities that threats exploit in these systems.

This report provides a taxonomy for machine learning algorithms, a detailed analysis of threats and security controls in widely adopted standards

Our analysis indicates that the conventional security controls, albeit very effective for information systems, need to be complemented by security controls tailored to machine learning functionalities. To identify these machine-learning controls, we conduct a systematic review of relevant literature, where academia and research institutes propose ways to avoid and mitigate threats targeting machine learning algorithms. Our report provides an extensive list of security controls that are applicable only for machine learning systems, such as “include adversarial examples to training datasets”. For all controls, we map the core functionality of machine learning algorithms that they intend to protect to the vulnerabilities that threats exploit.

Our findings indicate that there is no unique strategy in applying a specific set of security controls to protect machine learning algorithms. The overall cybersecurity posture of organisations who use machine learning algorithms can be enhanced by carefully choosing controls designed for these algorithms. As these controls are not validated in depth, nor standardised in how they should be implemented, further research should focus on creating benchmarks for their effectiveness. We further identified cases where the deployment of security controls may lead to trade-offs between security and performance. Therefore, the context in which controls are applied is crucial and next steps should focus on considering specific use cases and conducting targeted risk assessments to better understand these trade-offs. Finally, given the complexity of securing machine learning systems, governments and related institutions have new responsibilities in raising awareness regarding the impact of threats on machine learning. It is important to educate data scientists on the perils of threats and on the design of security controls before machine learning algorithms are used in organisations’ environments. By engaging experts in machine learning in cybersecurity issues, we may create the opportunity to design innovative security solutions and mitigate the emerging threats on machine learning systems.

1. INTRODUCTION

Artificial Intelligence (AI) has grown significantly in recent years and driven by computational advancements has found wide applicability. By providing new opportunities to solve decision-making problems intelligently and automatically, AI is being applied to more and more use cases in a growing number of sectors. The benefits of AI are significant and undeniable. However, the development of AI is also accompanied by new threats and challenges, which relevant professionals will have to face.

In 2020, ENISA published a threat landscape report on AI[1]. This report, published with the support of the Ad-Hoc Working Group on Artificial Intelligence Cybersecurity[2], presents the Agency's active mapping of the AI cybersecurity ecosystem and its threat landscape. This threat landscape not only lays the foundation for upcoming cybersecurity policy initiatives and technical guidelines, but also stresses relevant challenges.

Machine learning (ML), which can be defined as the ability for machines to learn from data to solve a task without being explicitly programmed to do so, is currently the most developed and promising subfield of AI for industrial and government infrastructures. It is also the most commonly used subfield of AI in our daily lives.

ML algorithms and their specificities, such as the fact that they need large amount of data to learn, make them the subject of very specific cyber threats that project teams must consider. The aim of this study is to help project teams identify the specific threats that can target ML algorithms, associated vulnerabilities, and security controls for addressing these vulnerabilities.

Building on the ENISA AI threat landscape mapping, this study focuses on cybersecurity threats specific to ML algorithms. Furthermore, vulnerabilities related to the aforementioned threats and importantly security controls and mitigation measures are proposed.

The adopted description of AI is a deliberate simplification of the state of the art regarding that vast and complex discipline with the intent of not precisely or comprehensively define it but rather pragmatically contextualise the specific technique of machine learning.

1.1 OBJECTIVES

The objectives of this publication are:

- To produce a taxonomy of ML techniques and core functionalities to establish a logical link between threats and security controls.
- To identify the threats targeting ML techniques and the vulnerabilities of ML algorithms, as well as the relevant security controls and how these are currently being used in the field to ensure minimisation of security risks.
- To propose recommendations on future steps to enhance cybersecurity in systems that rely on ML techniques.

1.2 METHODOLOGY

To produce this report, the work was divided into three stages. At the core of the methodology was an extensive literature review (full list of references may be found in Annex D). The aim

[1] https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges
[2] See https://www.enisa.europa.eu/topics/iot-and-smart-infrastructures/artificial_intelligence/ad-hoc-working-group/adhoc_wg_calls

was to consult documents that are more specific to ML algorithms in general in order to build the taxonomy, and to consult documents more specific to security to identify threats, vulnerabilities, and security controls. At the end of the systematic review, more than 200 different documents (of which a hundred are related to security) on various algorithms of ML had been collected and analysed.

First, we introduced a high-level **ML taxonomy**. To understand the vulnerabilities of different ML algorithms, how they can be threatened and protected, it is crucial to have an overview of their core functionalities and lifecycle. To do so, a first version of the desk research on ML-focussed sources was compiled and the ML lifecycle presented in ENISA's work on AI cybersecurity challenges was consulted[3]. We then analysed and synthesised all references to produce a first draft of the taxonomy. The draft was submitted and interviews were held with the ENISA Ad-Hoc Working Group on Artificial Intelligence Cybersecurity. After considering their feedback, the ML taxonomy and lifecycle were validated.

The second step was to identify the **cybersecurity threats that could target ML algorithms and potential vulnerabilities**. For this task, the threat landscape from ENISA's report on AI cybersecurity challenges was the starting point, which was then enriched through desk research with sources related to the security of ML algorithms. Additionally, the expertise of the ENISA Ad-Hoc Working Group on Artificial Intelligence Cybersecurity was sought. This work allowed us to select threats and identify associated vulnerabilities. Subsequently, they were linked to the previously established ML taxonomy.

The last step of this work was the identification of the **security controls** addressing the vulnerabilities. To do this, we utilised the desk research and enriched it with the most relevant standard security controls from ISO 27001/2 and the NIST 800-53 framework. The output was reviewed with the experts of the ENISA Ad-Hoc Working Group on Artificial Intelligence Cybersecurity. This work allowed us to identify security controls that were then linked to the ML taxonomy.

It is important to note that we opted to enrich the ML-targeted security controls with more conventional ones to highlight that applications using ML must also comply with more classic controls in order to be sufficiently protected. Considering measures that are specific to ML would only give a partial picture of the security work needed on these applications.

1.3 TARGET AUDIENCE

The target audience of this report can be divided into the following categories:

- **Public/governmental sector** (EU institutions and agencies, Member States' regulatory bodies, supervisory authorities in the field of data protection, military and intelligence agencies, law enforcement community, international organisations, and national cybersecurity authorities): to help them with their risk analysis, identify threats and understand how to secure ML algorithms.
- **Industry** (including Small and Medium Enterprises (SMEs)) that makes use of AI solutions and/or is engaged in cybersecurity, including operators of essential services: to help them with their risk analysis, identify threats and understand how to secure ML algorithms.
- **AI technical community, AI cybersecurity experts and AI experts** (designers, developers, ML experts, data scientists, etc.) with an interest in developing secure solutions and in integrating security and privacy by design in their solutions.
- **Cybersecurity community:** to identify threats and security controls that can apply to ML algorithms.

[3] https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges

- **Academia and research community**: to obtain knowledge on the topic of securing ML algorithms and identify existing work in the field.
- **Standardisation bodies**: to help identify key aspects to consider regarding securing ML algorithms.

1.4 STRUCTURE

The report aims to help the target audience to identify the cyber threats to consider and the security controls to deploy in order to secure their ML applications. Accordingly, the report is structure into three sections:

- **ML algorithms taxonomy**: first, a taxonomy to describe the main characteristics of the algorithms is defined. The different ML algorithms are categorised based on their core functionalities (e.g., the learning paradigm) and the lifecycle of a ML algorithm is defined.
- **Identification of relevant threats and vulnerabilities**: secondly, a list of the cybersecurity threats and associated vulnerabilities to consider for ML algorithms is defined. Threats are mapped to the taxonomy to highlight the link between them, the core functionalities, and the lifecycle of the ML algorithms.
- **Security controls**: thirdly, a list of security controls for addressing the previously considered vulnerabilities is given. They are also mapped to the ML taxonomy.

This report focuses on threats that target ML algorithms and on the associated security controls. It is important to note that this publication examines security controls that are specific to ML algorithms as well as standard security controls that are also applicable to ML algorithms and systems making use of them. To use this publication effectively, it is important to note that:

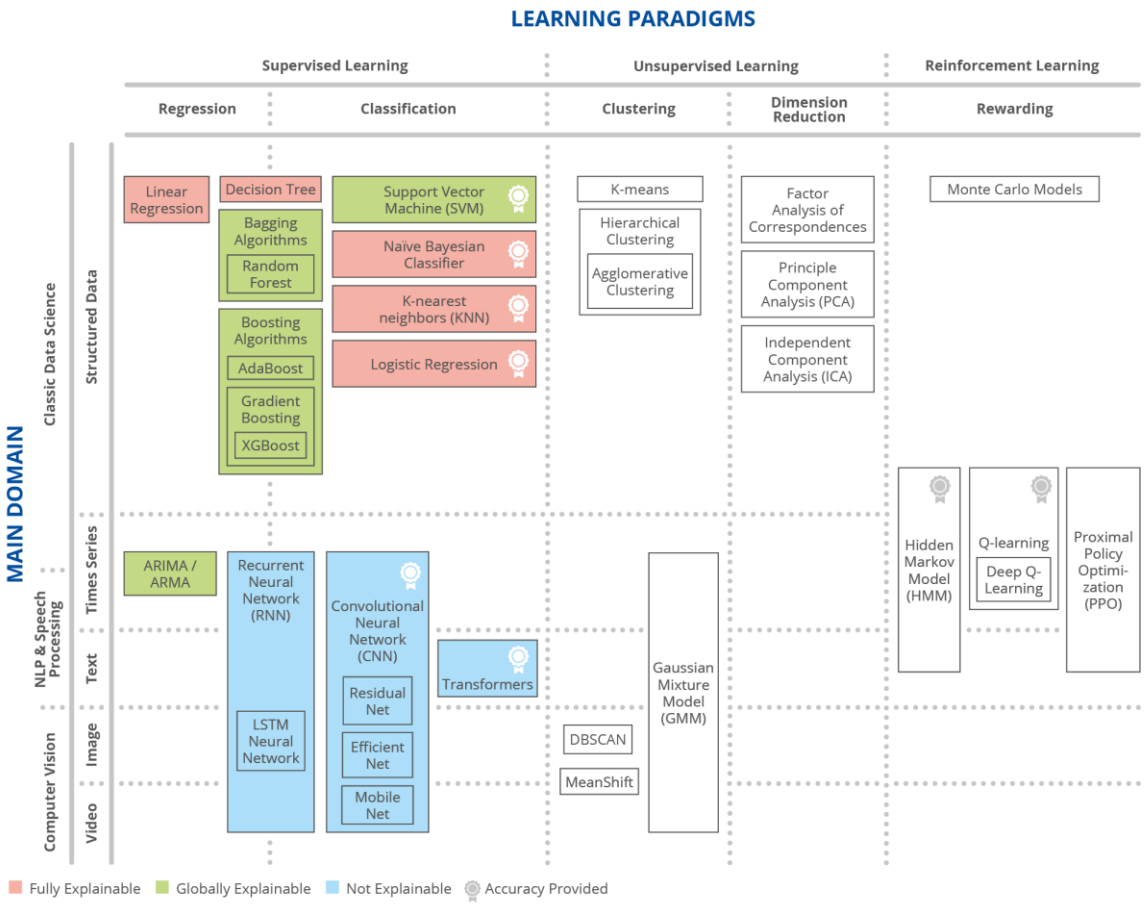
- As is the case for any application, when using ML, one must also consider traditional security standards (e.g. ISO 27001/2, NIST 800-53), because ML applications are subject not only to AI/ML specific threats but also to general nature cybersecurity threats.
- The context of the application (e.g. manipulated data, business case, deployment) must be considered to correctly assess the risks and prioritise deployment of the security controls accordingly.

2. MACHINE LEARNING ALGORITHMS TAXONOMY

One of the objectives of this work was to devise a (non-exhaustive) taxonomy, to support the process of identifying which specific threats can target ML algorithms, their associated vulnerabilities, and security controls for addressing these vulnerabilities. An important disclaimer needs to be made concerning this taxonomy, namely that it is not meant to be complete or exhaustive when it comes to ML, instead it aims to support the security analysis of ML algorithms in this report.

Based on the desk research and interviews with experts of the ENISA AI Working group, we identified 40 of the most commonly used ML algorithms. A taxonomy was built based on the analysis of these algorithms. In particular, it was noted that ML algorithms were driven mainly by the learning paradigms and the problem they address (main domain). These aspects were therefore chosen to form the key taxonomy dimensions, as seen in Figure 1. It should be noted that Annex A provides a complete listing of the 40 algorithms and their mapping to the features of the taxonomy, whereas the Figure serves for illustration purposes.

Figure 1: Machine Learning Algorithm taxonomy



There is a strong correlation between the domain of application (the problem being addressed) and the data type which is being worked on, as well as between data environments and learning paradigm. Thus, further dimensions of the taxonomy were introduced accordingly.

2.1 MAIN DOMAIN AND DATA TYPES

Different algorithms are used in different domains of ML. Therefore, the algorithms have been categorised according to the main domains represented. Three main domains were (non-exhaustively) selected, namely Computer Vision, NLP (Natural Language Processing) & Speech Processing (understanding and generating speech), and Classic Data Science.

The inputs that are given to a ML algorithm are data and therefore, the algorithms can be categorised based on the types of data that is fed into them. In most cases, specific types of data are used in certain domains of ML. Indeed, all the algorithms used in computer vision are fed with images and videos, in the same way that all algorithms used in Natural Language Processing are fed with text[4]. In Table 1, the main domains and the type of data used in each of them are listed.

Table 1: Main domains and data types

Main domain	Data type	Definition
Computer Vision	Image	Visual representation of a matrix of pixels constituted of 1 channel for black and white images, 3 elements (RGB) for coloured images or 4 elements (RGBA) for coloured images with opacity.
	Video	A succession of images (frames), sometimes grouped with a time series (a sound).
NLP & Speech processing	Text	A succession of characters (e.g. a tweet, a text field).
	Time series[5]	A series of data points (e.g. numerical) indexed in time order.
Classic Data Science	Structured Data	Data organised in a predefined model of array with one specific column for each feature (e.g. textual, numerical data, date). To be more accurate, structured data refer to organised data that can be found in a relational data base for example (that may contain textual columns as mentioned). Quantitative data can be distinguished from qualitative data. Quantitative data corresponds to the numerical data that can supports some arithmetic operations whereas qualitative data is usually used as categorical data to classify data according to their similarities.

Certain domains such as NLP and Computer Vision have been separated from Classic Data Science. The purpose of this separation was to make a distinction between algorithms that may be used specifically or predominantly for each domain.

[4] Audio data are also used for speech recognition. For the purposes of this report, we consider only text for the NLP for the taxonomy. considering that this will not create differences for the work on threats.
[5] For the purposes of this report, time series belong to the two main domains: Classic Data Science and Speech processing. By restraining Time series to Classic Data Science and Speech processing, we aspired to emphasise the specific approaches that are used for this domain like ARIMA and Hidden Markov Model. Furthermore, we include audio data under time series and made the choice to separate video from time series.

2.2 LEARNING PARADIGMS

Learning paradigm in ML relates to how a machine learns when data is fed to it. For example, all the classification and regression algorithms use labelled data, meaning that they are doing only supervised learning. Indeed, supervised learning, by definition, is the learning of labelled data, which can be either numerical (in this case, the learning paradigm is regression), or categorical (the learning paradigm is classification). An example of classification can be differentiating a cat from a dog in a picture, and an example of regression can be predicting the price of a house. On the other hand, a clustering algorithm uses unlabeled data, which is an unsupervised type of learning. Therefore, one can conclude that each learning paradigm is a specific case of one data environment.

In addition to the data types fed into the algorithms, we also focused on three learning paradigms, namely supervised learning, unsupervised learning, and reinforcement learning:

- Supervised learning learns a function that maps an input to an output based on example input-output pairs. It infers a function from labelled training data consisting of a set of training examples.
- Unsupervised learning learns patterns from unlabelled data. It discovers hidden patterns or data groupings without the need for human intervention.
- Reinforcement learning enables an agent to learn in an interactive environment by trial and error using feedback from its own actions and experiences.

Table 2: Learning paradigms with typical subtypes.

Learning paradigm	Subtypes	Definition
Supervised learning	Classification	Classification is the process of predicting the class of given data points. (Is the picture a cat or a dog?)
	Regression	Regression models are used to predict a continuous value. (Predict the price of a house based on its features).
Unsupervised learning	Clustering	Clustering is the task of dividing a set of data points into several groups such that data points in the same groups are more similar each other than from the data points of the other groups.
	Dimensionality reduction	Dimensionality reduction refers to techniques for reducing the number of input variables in training data.
Reinforcement learning	Rewarding	Rewarding is an area of ML concerned with how intelligent agents ought to take actions in an environment to maximise the notion of cumulative reward, learning by using feedback from their experiences.

Each of these learning paradigms have different security-related properties which may lead to attacks and therefore, it is relevant to represent this information in the taxonomy of ML algorithms, from which security controls will be mapped. For instance, the most common learning paradigm is classification and thus, it has many more examples of vulnerabilities due to its popularity.

2.3 NAVIGATING THE TAXONOMY

Each algorithm is placed in its corresponding cell of the taxonomy grid, according to its learning paradigm, data type and main domain. For instance, Recurrent Neural Networks[6] (RNN), which are a type of neural network helpful in modelling sequenced data, are used for regression in supervised learning, so they must be mapped in the first column. Moreover, the data fed into them can be text, time series, images, or videos so the RNN box covers all the corresponding lines in the taxonomy.

However, some of the widely used and mentioned algorithms are based on common elementary components, or are extensions of the same principle, and can therefore form families or clusters of algorithms on this taxonomy grid. Hence, we map those specific algorithms in groups by using nested boxes, as it allows for the representation of a wide variety of algorithms, while showing that some have relationships with one another.

To continue with the previous example, a more recent version of RNN is LSTM[7] (Long-Short Term Memory), which differs from RNN based on its optimisation techniques, making it faster to learn and more precise. Since LSTM is a specific extension of RNN, the LSTM box was nested in the RNN box in the taxonomy: this indicates that the two algorithms are part of the same family.

2.4 EXPLAINABILITY AND ACCURACY

An important aspect of security of AI is that of explainability. Understanding the algorithms and making them explainable makes them more accessible to as many people as possible. It also helps to increase the trustworthiness of AI and support forensics and analysis of decisions. Following inputs from the desk research exercise and from the research on attacks targeting ML models, we additionally included two important parameters in the taxonomy:

- **Explainability**: For the purposes of this study, algorithms are deemed to be "explainable" if the decision it makes can be understood by a human. That is to say, decisions can be understood by a human such as a developer or an auditor and then explained to an end-user, for example. To be fully explainable, an algorithm must be:
    - Globally explainable: a user can identify the features' importance for the trained model.
    - Locally explainable: a user can explain why the algorithm gives a specific output (prediction) to a specific input data (features' values).

- **Accuracy (probability score)**: Some algorithms provide, in addition to a predictive output, the probability of this prediction which can be interpreted as an "accuracy level". If an algorithm doing classification predicts that a picture of a cat is indeed a picture of a cat at 95% accuracy, one can say that the algorithm has a "high accuracy classification". Otherwise, if the prediction was at 55% accuracy, one could say that the algorithm has a "low accuracy classification".

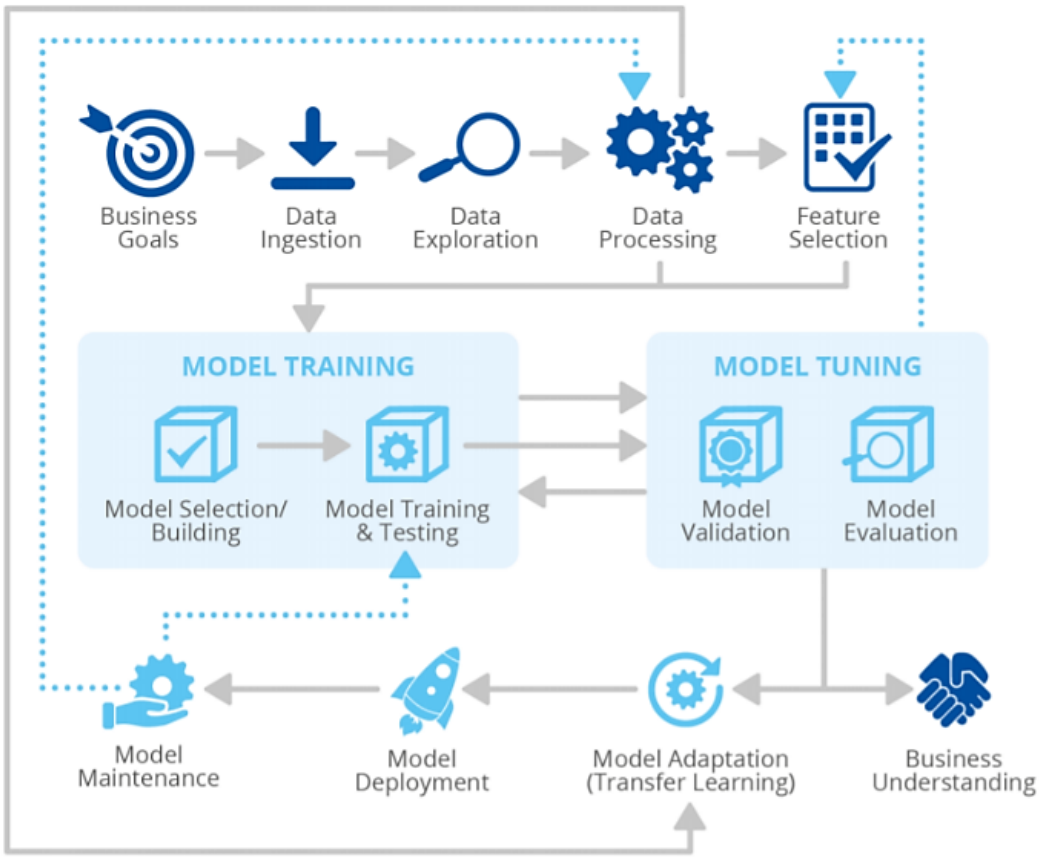
It is important to note that we focused on the algorithms' explainability because this work is important for other parts of the publication. For example, in one identified security control, it is highlighted that it is necessary to ensure that ML projects comply with regulatory constraints such as the GDPR, which describes some explainability requirements[8].

[6] https://apps.dtic.mil/dtic/tr/fulltext/u2/a164453.pdf
[7] https://www.bioinf.jku.at/publications/older/2604.pdf
[8] GDPR Recital 71 "The data subject should have the right not to be subject to a decision, which may include a measure, evaluating personal aspects relating to him or her which is based solely on automated processing and which produces legal effects concerning him or her or similarly significantly affects him or her, such as automatic refusal of an online credit application or e-recruiting practices without any human intervention. […] In any case, such processing should be subject to
[1] Please use footnotes for providing additional or explanatory information and/or relevant links. References should be listed in a dedicated section. Use only the function References/Insert Footnote

2.5 AN OVERVIEW OF END-TO-END MACHINE LEARNING LIFECYCLE

An ML system lifecycle includes several interdependent phases ranging from its design and development (including sub-phases such as requirement analysis, data collection, training, testing, integration), installation, deployment, operation, maintenance, and disposal. It defines the phases that an organisation should follow to take advantage of AI and of ML models in particular to derive practical business value. The latter can be represented as the architecture illustrated in Figure 22[9]:

Figure 2: Typical AI lifecycle (from the ENISA AI Threat Landscape)



Building on the AI lifecycle, we describe in Figure 3 an overview of a typical ML lifecycle with a complete overview of the principal steps.

suitable safeguards, which should include specific information to the data subject and the right to obtain human intervention, to express his or her point of view, to obtain an explanation of the decision reached after such assessment and to challenge the decision."

[9] https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges

[1] Please use footnotes for providing additional or explanatory information and/or relevant links. References should be listed in a dedicated section. Use only the function References/Insert Footnote

Figure 3: ML Algorithm lifecycle[10,11]

The aim of the ML algorithm taxonomy is to focus not only on the functionalities of the algorithms but also on the ML models' workflow represented by the lifecycle. This lifecycle summarises the principle steps to produce an ML model. It is important to note that several steps could have been added, such as data creation and data analysis (for instance, to analyse if there are some personal data or biases). However, to simplify the lifecycle, some steps have been condensed. Thus, for example, data cleaning has been included. Regarding data creation, it was considered as being external to the ML lifecycle.

[10] Optimisation is also known as model tuning.
[11] Data cleaning and data processing have been separated to distinguish the cleaning phase from the adaptation phase of the dataset for learning (dimension reduction, feature engineering, etc.).

3. ML THREATS AND VULNERABILITIES

3.1 IDENTIFICATION OF THREATS

Based on the methodology described in the Introduction and using a combination of desktop research and experts' interviews, we identified a list of six high-level threats and seven sub-threats that were then mapped to the taxonomy. It is important to note that:

- Threats against supporting infrastructures are not analysed in this publication.
- All threats relate to the previous ENISA publication on the AI Threat Landscape; accordingly, they have been mapped to AI assets (environments, tools, data, etc.)

The table following summarises Machine Learning threats and includes:

- Threats and sub-threats definitions.
- Whether they are specific to ML algorithms or not.
- At which stage of the life cycle defined in the first section the threat is likely to occur.

Table 3: Threats and sub-threats

Threats | *sub-threats*	Definition	Stage of the lifecycle								
		Data Collection	Data Cleaning	Data Preprocessing	Model design	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Evasion	A type of attack in which the attacker works on the ML algorithm's inputs to find small perturbations leading to large modification of its outputs (e.g. decision errors). It is as if the attacker created an optical illusion for the algorithm. Such modified inputs are often called adversarial examples. Example: the projection of images on a house could lead the algorithm of an autonomous car to take the decision to suddenly make it brake.										x
Use of adversarial examples crafted in white or grey box conditions (e.g. FGSM…)	In some cases, the attacker has access to information (model, model parameters, etc.) that can allow him to directly build adversarial examples. One example is to directly use the model's gradient to find the best perturbation to add to the input data to evade the model.										x
Oracle	A type of attack in which the attacker explores a model by providing a series of carefully crafted inputs and observing outputs. These attacks can be previous steps to more harmful types, evasion or poisoning for example. It is as if the attacker made the model talk to then better compromise it or to obtain information[204] about it (e.g. model extraction) or its training data (e.g. membership inferences attacks and Inversion attacks). Example: an attacker studies the set of input-output pairs and uses the results to retrieve training data.										x
Poisoning	A type of attack in which the attacker altered data or model to modify the ML algorithm's behavior in a chosen direction (e.g. to sabotage its results, to insert a backdoor). It is as if the attacker conditioned the algorithm according to its motivations. Such attacks are also called causative attacks. Example: massively indicating to an image recognition algorithm that images of dogs are indeed cats to lead it to interpret it this way.	x	x	x	x	x		x		x	x
Label modification	An attack in which the attacker corrupts the labels of training data. This sub-threat is specific to Supervised Learning.	x	x	x		x					
Model or data disclosure	This threat refers to the possibility of leakage of all or partial information about the model.[12] Example: the outputs of a ML algorithm are so verbose that they give information about its configuration (or leakage of sensitive data)	x	x	x	x	x	x	x	x	x	x

[12] We have chosen to separate the oracle attacks from this threat to address the specifics of both threats and give them both a fair representation. However, Oracle-type attacks may be considered as a ML specific sub-threat of model or data disclosure.

Threats | *sub-threats*	Definition	Stage of the lifecycle								
		Data Collection	Data Cleaning	Data Preprocessing	Model design	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Data disclosure	This threat refers to a leak of data manipulated by ML algorithms. This data leakage can be explained by an inadequate access control, a handling error of the project team or simply because sometimes the entity that owns the model and the entity that owns the data are distinct. To train the model, it is often necessary for the data to be accessed by the model provider. This involve sharing the data and thus share sensitive data with a third party.	x	x	x	x	x	x	x	x	x	x
Model disclosure	This threat refers to a leak of the internals (i.e. parameter values) of the ML model. This model leakage could occur because of human error or contraction with a third party with a too low security level.				x	x	x	x	x	x	x
Compromise of ML application components	This threat refers to the compromise of a component or developing tool of the ML application. Example: compromise of one of the open-source libraries used by the developers to implement the ML algorithm.	x	x	x	x	x	x	x	x	x	x
Failure or malfunction of ML application	This threat refers to ML application failure (e.g. denial of service due to bad input, unavailability due to a handling error). Example: the service level of the support infrastructure of the ML application hosted by a third party is too low compared to the business needs, the application is regularly unavailable. Note that this threat does not consider failure of business use cases (for example, the algorithm fails because it is not accurate enough to handle all real-life situations it is exposed to).									x	x
Human error	The different stakeholders of the model can make mistakes that result in a failure or malfunction of ML application. For example, due to lack of documentation, they may use the application in use-cases not initially foreseen.	x	x	x	x	x	x	x	x	x	x
Denial of service due to inconsistent data or a sponge example	ML algorithms usually consider input data in a defined format to make their predictions. Thus, a denial of service could be caused by input data whose format is inappropriate. It may also happen that a malicious user of the model constructs an input data (a sponge example) specifically designed to increase the computation time of the model and thus potentially cause a denial of service.										x
Cybersecurity incident not reported to incident response teams	This threat refers to the possibility that a project team may not report security incidents to dedicated teams while a policy of mandatory incident reporting has been defined.	x	x	x	x	x	x	x	x	x	x

3.2 VULNERABILITIES MAPPED TO THREATS

To identify the security controls, we determined vulnerabilities associated with the threats described in the previous section. It is important to note that the same vulnerabilities may be found behind one or more threats (e.g. the “Poor access management” vulnerability). The table below lists vulnerabilities of ML algorithms and maps them to the aforementioned threats.

Table 4: Threats and associated vulnerabilities

Threats | *sub-threats*		Vulnerabilities
Evasion		Lack of detection of abnormal inputs
		Poor consideration of evasion attacks in the model design implementation
		Poor consideration of evasion attacks in the model design implementation
		Lack of training based on adversarial attacks
		Using a widely known model allowing the attacker to study it
		Inputs totally controlled by the attacker which allows for input-output-pairs
	Use of adversarial examples crafted in white or grey box conditions (e.g. FGSM…)	Too much information available on the model
		Too much information about the model given in its outputs
Oracle		Poor access rights management
		The model allows private information to be retrieved
		Too much information about the model given in its outputs
		Too much information available on the model
		Lack of consideration of attacks to which ML applications could be exposed to
		Lack of security process to maintain a good security level of the components of the ML application
		Weak access protection mechanisms for ML model components
Poisoning		Model easy to poison
		Lack of data for increasing robustness to poisoning
		Poor access rights management
		Poor data management
		Undefined indicators of proper functioning, making complex compromise identification
		Lack of consideration of attacks to which ML applications could be exposed to
		Use of uncontrolled data
		Use of unsafe data or models (e.g. with transfer learning)
		Lack of control for poisoning
		No detection of poisoned samples in the training dataset
		Weak access protection mechanisms for ML model components
	Label modification	Use of unreliable sources to label data
Model or data disclosure		Poor access rights management
		Existence of unidentified disclosure scenarios
		Weak access protection mechanisms for ML model components

Threats | *sub-threats*		Vulnerabilities
		Lack of security process to maintain a good security level of the components of the ML application
		Unprotected sensitive data on test environments
	Data disclosure	Too much information about the model given in its outputs
		The model can allow private information to be retrieved
		Disclosure of sensitive data for ML algorithm training
	Model disclosure	Too much information available on the model
		Too much information about the model given in its outputs
Compromise of ML application components		Poor access rights management
		Too much information available on the model
		Existence of several vulnerabilities because the ML application was not included into process for integrating security into projects
		Use of vulnerable components (among the whole supply chain)
		Too much information about the model given in its outputs
		Existence of unidentified compromise scenarios
		Undefined indicators of proper functioning, making complex compromise identification
		Bad practices due to a lack of cybersecurity awareness
		Lack of security process to maintain a good security level of the components of the ML application
		Weak access protection mechanisms for ML model components
		Existence of several vulnerabilities because ML specificities are not integrated to existing policies
		Existence of several vulnerabilities because ML application do not comply with security policies
		Contract with a low security third party
Failure or malfunction of ML application		Existing biases in the ML model or in the data
		ML application not integrated in the cyber-resilience strategy
		Existence of unidentified failure scenarios
		Undefined indicators of proper functioning, making complex malfunction identification
		Lack of explainability and traceability of decisions taken
		Lack of security process to maintain a good security level of the components of the ML application
		Existence of several vulnerabilities because ML specificities are not integrated in existing policies
		Contract with a low security third party
		Application not compliant with applicable regulations
	Human error	Poor access rights management
		Lack of documentation on the ML application
	Denial of service due to inconsistent data or a sponge example	Use of uncontrolled data
	Cybersecurity incident not reported to incident response teams	Lack of cybersecurity awareness

4. SECURITY CONTROLS

4.1 SECURITY CONTROLS RESULTS

Having identified a set of threats that can target vulnerabilities in applications which use ML algorithms, it is possible to identify which security controls can be put in place to mitigate them. To do this, we commenced with the vulnerabilities identified in the previous Chapter and came up with a list of 37 security controls that were then mapped to the taxonomy. Table 5 summarises security controls for ML algorithms and lists:

- Security controls definitions.
- At which stage of the lifecycle the security controls can be applied.

For ease of reading, they were divided into three categories:

- "Organisational and Policy" are more traditional security controls, either organisational or linked to security policies.
- "Technical" are more classic technical security controls.
- "Specific to ML" are security controls that are specific to applications using ML.

In Annex 5.C, a set of operational implementation examples are listed for each of the security controls. This includes:

- For security controls not specific to ML algorithms: examples from the ISO 27001/2[13] family of standards or NIST 800-53[14] framework that should be considered when implementing the security control.
- For security controls specific to ML: examples of techniques found in the current literature. All sources are referenced and may be found in Annex 5.D.

The overall mapping of threats, vulnerabilities and security controls is available in Annex 5.B.

[13] https://www.iso.org/isoiec-27001-information-security.html
[14] https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

Table 5: Security controls

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
ORGANISATIONAL											
Apply a RBAC model, respecting the least privileged principle	Define access rights management using a RBAC (Role Based Access Control) model respecting the least privileged principle. This should cover all components of the ML model (e.g. host infrastructures) and allow for the protection of resources such as the model (e.g. its configuration, its code) and the data it used (e.g. training data). It is notable that the roles to be included also concern the end user. For example: the end user who can submit inputs to the model should not be able to have access to its configuration.	x	x	x	x	x	x	x	x	x	x
Apply documentation requirements to AI projects	As for all projects, documentation must be produced for AI to preserve knowledge on the choices made during the project phase, the application architecture, its configuration, its maintenance, how to maintain its effectiveness over time and the assumptions made about the model use. This documentation should also include the changes that will be applied, including to the documentation throughout the algorithm's life cycle.	x	x	x	x	x	x	x	x	x	x
Assess the regulations and laws the ML application must comply with	As all applications, those using ML can be subject to regulations and laws (e.g., depending on collected data). Such assessment must be done as soon as possible during the project phase, and should be regularly updated thereafter as regulations are rapidly evolving (e.g., an AI Act has been proposed at the European level).	x	x	x	x	x	x	x	x	x	x
Ensure ML applications comply with data security requirements	As all applications, those using ML must comply with data security requirements to ensure the overall lifecycle of the data they use will be secured (e.g. description of data lifecycle and associated controls, data classification, protection of data at rest and in transit, use of appropriate cryptographic means, data quality controls).	x	x	x	x	x	x	x	x	x	x
Ensure ML applications comply with identity management, authentication, and access control policies	As all applications, those using ML must comply with defined policies regarding identity management (e.g. ensure all users are integrated in the departure process), authentication (e.g. passwords complexity, use of Multi-Factors Authentication (MFA), access restriction) and access control (e.g. RBAC model, connection context). Underlying security requirements must be applied to all ML application components (e.g. model configuration, host infrastructures, training data).	x	x	x	x	x	x	x	x	x	x

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Ensure ML applications comply with protection policies and are integrated to security operations processes	As all applications, those using ML must comply with protection policies (e.g. hardening, anti-malware policy) and be integrated to security operations processes (e.g. vulnerability management, backups).	x	x	x	x	x	x	x	x	x	x
Ensure ML applications comply with security policies	As all applications, those using ML must comply with existing security policies.	x	x	x	x	x	x	x	x	x	x
Include ML applications into detection and response to security incident processes[15]	As all applications, those using ML must be integrated in global processes for detection and incident response. This implies collecting the appropriate logs, configuring relevant detection use cases to detect attacks on the application, and giving the keys to incident response team for efficient response.	x	x	x	x	x	x	x	x	x	X
Include ML applications in asset management processes	As all applications, those using ML must be integrated to global processes for asset management to ensure their assets are inventoried, their owners are identified, their information classified.	x	x	x	x	x	x	x	x	x	X
Integrate ML applications into the overall cyber-resilience strategy	As any application, ML ones must be integrated in the overall cyber-resilience strategy, to ensure their architecture and operational processes (e.g. backups) take into account cybersecurity scenario.	x	x	x	x	x	x	x	x	x	x
Integrate ML specificities to existing security policies	Specific ML security attention points should be integrated in existing security policies and guidelines to ensure they are taken into consideration.	x	x	x	x	x	x	x	x	x	x
TECHNICAL											
Assess the exposure level of the model used	Some model designs are more commonly used or shared than others and, especially in the ML field; it can be included in their lifecycle to widely share them (e.g. open source sharing). These aspects must be considered in the global application risk analysis. For example, two elements can be distinguished: - Do not reuse models taken directly from the internet without checking them. - Use models for which the threats are clearly identified and for which security controls exist.	x	x	x	x	x	x	x	x	x	x

[15] Please note that ML components with false positives might have adverse effect.

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Check the vulnerabilities of the components used so that they have an appropriate security level	During the lifecycle of an ML algorithm, several components (such as software, programming libraries or even other models) are used to complete the project. Security checks have to be carried out to ensure that these components offer an adequate level of security. Moreover, some mechanisms need to be used to prevent tampering with the components used. For example: if an open-source library is to be used, code reviews or check for public vulnerabilities on it can be done.	x	x	x	x	x	x	x	x	x	x
Conduct a risk analysis of the ML application	A risk analysis of the overall application should be conducted to take into account the specificities of its context, including: - The attacker's motivations - The sensitivity of the data handled (e.g. medical or personal and thus subject to regulatory constraints, strategic for the company and should thus be highly protected) - The application hosting (e.g. through third parties services, cloud or on premise environments) - The model architecture (e.g. its exposition, learning methods) - The ML application lifecycle (e.g., model sharing	x	x	x	x	x	x	x	x	x	x
Control all data used by the ML model	Data must be checked to ensure they will suit the model and limit the ingestion of malicious data: - Evaluate the trust level of the sources to check it's appropriate in the context of the application - Protect their integrity along the whole data supply chain - Their format and consistence are verified - Their content is checked for anomalies, automatically or manually (e.g. selective human control) - In the case of labeled data, the issuer of the label is trusted.	x	x	x	x	x	x	x	x	x	x
Ensure reliable sources are used	ML is a field in which the use of open-source elements is widespread (e.g., data for training, including labeled ones, models). The trust level of the different sources used should be assessed to prevent using compromise ones. For example: the project wants to use labeled images from a public library. Are the contributors sufficiently trusted to have confidence in the contained images or the quality of their labelling?	x			x						
Use methods to clean the training dataset from suspicious samples	Removing suspicious samples from the training and testing dataset can help prevent poisoning attacks. Some methods exist to identify those that could cause strange behavior of the algorithm.		x	x		x					

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Define and monitor indicators for proper functioning of the model	Define dashboards of key indicators integrating security indicators (peaks of change in model behavior etc.) to follow-up the proper functioning of the model regarding the business case, in particular to allow rapid identification of anomalies.										x
Ensure appropriate protection is deployed for test environments	Test environments must also be secured according to the sensitivity of the information they contain. Special care must be paid to the data used in these environments, to ensure their protection (e.g., same protection measures as for production if not desensitiser).	x	x	x	x	x	x	x	x	x	x
Ensure ML applications comply with third parties' security requirements	As all applications, those using ML must comply with third parties' security requirements if their context involves suppliers.	x	x	x	x	x	x	x	x	x	x
Ensure ML projects follow the global process for integrating security into projects	As any project, ML projects must comply to process for integrating security into projects, including the followings: - Risk analysis on the whole application - Check of the integration of cybersecurity best practices regarding architecture, secure development. - Check that the application will be integrated in existing operational security processes: monitoring and response, patch management, access management, cyber-resilience. - Check of the production of adequate documentation to ensure the sustainability of the application (e.g., technical architecture, hardening, exploitation, configuration and installation documents) - Security checks before going to production (e.g. security audit, pen tests)	x	x	x	x	x	x	x	x	x	x
SPECIFIC ML											
Add some adversarial examples to the training dataset[16]	Include adversarial examples to the algorithm's training to enable it to be more resilient to such attacks. Depending on the application domain and ambient conditions, such training could be done continuously.	x	x	x							
Apply modifications on inputs[17]	Adding a step to modify the model's inputs (e.g. data randomisation which consists in adding random noise to each piece of data), can improve the robustness of the model to attacks. Such steps can make it more difficult for an attacker to understand the functioning of the algorithm and thus to manipulate it and reduce the impacts of an attack. This security control can be applied during training or model deployment stages.			x							x

[16] This security control is often referred to as "Robust adversarial training" in the literature.
[17] One important thing to keep in mind is that such modifications should not overly impact model performance on benign inputs.

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Build explainable models	The ML models should be explainable, even if it means simplifying them, to enable a good understanding of their functioning and decision factors. It can also be a regulatory requirement (e.g. GDPR). However, once again, security interferes with the explainability property of the model (easier-to-understand decisions can be easier-to-build adversarial examples). It is therefore a trade-off between the need for explainability and security.				x				x		
Choose and define a more resilient model design	Some model designs can be more robust than others against attacks. For instance, ensemble methods like bagging can mitigate the impact of poisoning (during the training phase). Another example is defensive distillation, which may allow deep neural networks to better deal with evasion attacks.				x						
Enlarge the training dataset	Using a set of training data expansion techniques (e.g. data augmentation) addresses the lack of data and improves the robustness of the model to poisoning attacks by diluting their impact. It is notable, however, that this security control more specifically addresses poisoning attacks that aim to reduce the performance of the model than those that seek to establish a backdoor. Moreover, one needs to ensure the reliability of the sources used to augment the dataset.	x		x							
Ensure that models are unbiased	The introduction of bias in ML algorithms will not be detailed because it is not the topic of the publication. However, some techniques can be used to mitigate bias: verify the training dataset is representative enough regarding the business case, check the relevance of the attributes used to make decisions etc.	x	x	x	x	x			x		
Ensure that models respect differential privacy to a sufficient degree	Differential privacy (DP) is a strong, mathematical definition of privacy in the context of statistical and ML analysis. According to this mathematical definition, DP is a criterion of privacy protection, which many tools for analysing sensitive personal information have been devised to satisfy. It is noticeable that this security control can greatly reduce the performance of the model. It is therefore important to estimate the need for data or model protection. Example: Differential privacy makes it possible for technology companies to collect and share aggregate information about user habits, while maintaining the privacy of individual users.		x	x	x	x	x	x	x		x
Ensure that the model is sufficiently resilient to the environment in which it will operate.	Ensure that the model is sufficiently resilient against the environment in which it will operate. This includes, for instance, ensure that learning process and data are representative enough of the real conditions in which the model will evolve**.**	x	x	x	x	x	x	x	x	x	X

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Implement processes to maintain security levels of ML components over time	ML is a rapidly evolving field, especially regarding its cybersecurity. Regular checking of new attacks and defenses must be integrated into the processes for maintaining security level applications. The security level should thus be regularly assessed too.	x	x	x	x	x	x	x	x	x	X
Implement tools to detect if a data point is an adversarial example or not	Input-based detection tools can be of interest to identify whether a given input has been modified by an attacker or not. One example, in the case of Deep Neural Networks (DNNs), is to add a neural subnetwork to an architecture trained to detect adversarial examples.				x	x					x
Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	ML considerations should be added to awareness programs for concerned stakeholders and they must all receive cybersecurity awareness training: - Global cybersecurity awareness training including best practices to prevent attackers compromising the ML application. - Manipulation of potentially sensitive data or data subject to regulatory restrictions. - Configurations to prevent applications being vulnerable - ML-specific attack awareness	x	x	x	x	x	x	x	x	x	x
Integrate poisoning control after the "model evaluation" phase	Before moving the model to production and then on a regular basis, the model should be evaluated to ensure it has not been poisoned. This differs from the security control "*Use methods to clean the training dataset from suspicious samples*". Indeed, here, it's the model itself that is evaluated. For example: deep learning classification algorithms can be checked for poisoning using the STRIP[18] technique. The principle is to disturb the inputs and observe the randomness of the predictions.								x		
Reduce the available information about the model	This defense consists of limiting the information about the model when it is not necessary. More precisely, it aims at taking the necessary actions in order to reduce the information available on the model such as information on the training data set or any other information that could be used by an attacker (e.g., not publishing the model in open source). Of course, there is a trade-off between security and the fact that stakeholders (e.g., users, ML teams) sometimes want open source models. However, it remains notable that in many cases, research has shown that minimal information is sufficient to mount attacks.	x	x	x	x	x	x	x	x	x	x

[18] See https://arxiv.org/pdf/1902.06531.pdf. It is notable that STRIP (STRong Intentional Perturbatio) may have a huge runtime overhead and may be infeasible for large dataset.

Security controls	Definition	Stages of the lifecycle									
		Data Collection	Data Cleaning	Data Preprocessing	Model design and Implementation	Model Training	Model Testing	Optimisation	Model Evaluation	Model Deployment	Monitoring
Reduce the information given by the model[19]	Controlling the information (like its verbosity) provided by the model by applying basic cybersecurity hygiene rules is a way of limiting the techniques that an attacker can use to build adversarial examples. One of the basic rules of hygiene, for example, is to reduce the information of the output determined by the model to the maximum, or by profile making the request. For example: considering a classification application, it would consist of communicating only the predicted class to the users of solution, not the associated probability. However, it remains notable that in many cases, research has shown that minimal information is sufficient to mount attacks.										x
Use federated learning to minimize risk of data breaches	Federated learning is a set of training techniques that trains a model on several decentraliser servers containing local data samples, without exchanging their data samples. This avoids the need to transfer the data and/or entrust it to an untrusted third party and thus helps to preserve the privacy of the data.				x	x					
Use less easily transferable models[20]	The transferability property can be used to force adversarial examples from a substitution model to evade another. The ease of transferring an adversarial example from a model to another depends on the family of algorithms. One possible defense is thus to choose an algorithm family that is less sensitive to the transferability of adversarial examples.				x						

[19] It is important to keep in mind that, in case of attacks like evasion or oracle, this security control can help. However, in some cases, it may possible to bypass the security control by using more queries.
[20] Some evasion attacks are based on the following principle: train a model with data like the target model used and generate adversarial examples from this model. Then, present these adversarial examples to the target model to perform an evasion attack. Whether or not to transfer an adversarial example generated by one model to another depends on their respective design as shown in the reference 215.

5. CONCLUSION

Machine Learning algorithms are at the core of modern AI systems and applications. However, they are faced with a series of threats and vulnerabilities. In this report we have identified multiple security controls that can be applied to ML applications to address the threats they face. Some of the security controls are specific to ML algorithms, but others are standard technical and organisational cybersecurity controls to mitigate general attacks. It is important to apply both types of controls because AI systems, in addition to ML specific vulnerabilities, there exist also general type of vulnerabilities, which may also be exploited by adversaries.

There is no silver bullet for ML-specific attacks, but mitigation measures can still raise the bar for attackers. Thus, more attention should be given to security controls to enable comparability and increase resilience.

Mitigation controls for ML-specific attacks outlined in the report should in general be deployed during the entire lifecycle of the ML system. This includes measures for assuring the data quality and protecting its integrity, making the ML algorithms more robust and controlling access to both the model and the data to ensure their privacy. The report also emphasizes the need for the explainability of decisions, and the importance of detecting bias that can be present or injected in a model by an attacker, which can then lead to unethical uses of AI.

An important point highlighted in the report is that the identified security measures can be applied to all algorithms. Nevertheless, their operational implementations (see Annex C) may be specific to certain types of algorithms. For example, for the security control “Choose and define a more resilient model design”, the defensive distillation implementation is specific to neural networks. It is also notable that with the prevalence of research papers on supervised learning, there are more examples of operational implementations for this type of algorithms.

This report addresses an emerging subject. Thus, it remains very important to keep an active watch on threats and security controls in the field of ML in order to understand the latest innovations both from a technical point of view, or with a view to comply with standards provided by ISO, IEEE and ETSI[21].

When looking ahead and given the complexity of the issue of securing ML, companies and governments have new responsibilities. For instance, it is increasingly important to raise cybersecurity awareness within companies, especially regarding the security of ML systems. For some populations, particularly data science teams, cybersecurity has not been at the forefront for many years. Moreover, by including data science actors in these actions, they are also given the opportunity to think of innovative solutions to mitigate the various threats. Thus, to this end, training and education programs should be organised regularly and the vulnerabilities of ML should be demonstrated using concrete examples.

Finally, the context in which security controls are applied is crucial and specific use cases should be considered when conducting targeted risk assessments. All mitigations used should be proportional to the application-specific threat level and consider specific conditions of the environment that may either favor or hamper attacks. Moreover, defenders should be aware of the following points:

1) There is no silver bullet for mitigating ML-specific attacks. Some security controls may be bypassed by adaptive attackers. However, applied mitigations can still raise the bar for attackers.

[21] https://www.etsi.org/committee/sai

2) ML-specific mitigation controls are not generally evaluated in a standardised way even if it is a current and important issue to enable comparability. More research should be devoted to standardised benchmarks for comparing ML-specific mitigations on a level playing field. These benchmarks should also be enforced to ensure that the methods used in practice are the ones that perform best.

3) Deploying security controls often leads to a trade-off between security and performance and this is a topic of particular importance that should be further pursued by the research and cybersecurity communities.

A ANNEX: TAXONOMY OF ALGORITHMS

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
AdaBoost	AdaBoost uses multiple iterations to generate a single composite strong learner by iteratively adding weak learners. During each phase of training, a new weak learner is added to the ensemble, and a weighting vector is adjusted to focus on examples that were misclassified in previous rounds.	Classic Data Science	Structured data	Supervised learning	Classification, Regression	Globally Explainable		38
Adam optimisation	Adam optimisation is an extension to Stochastic gradient decent and can be used in place of classical stochastic gradient descent to update network weights more efficiently, thanks to two methods: adaptative learning rate and momentum	Classic Data Science	Structured data	/	Optimisation			24
Agglomerative clustering	Agglomerative clustering is a "bottom-up" approach of hierarchical clustering. Each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.	Classic Data Science	Structured data	Unsupervised Learning	Clustering			32
ARMA/ARIMA model	Given a time series Xt, the ARMA/ARIMA model is a tool to understand and predict the future values of this series. The model is composed of two parts: an autoregressive part (AR) and a moving average part (MA)	Classic Data Science	Time series	Supervised learning	Regression	Fully Explainable		136
BERT	Bidirectional Encoder Representations from Transformers (BERT) is a Transformer-based ML technique for natural language processing (NLP) pre-training developed by Google.	NLP & Speech processing	Text	Supervised learning	Classification	Not Explainable	Yes	5
Convolutional Neural Network	A Convolutional Neural Network is a deep learning algorithm which can take in an input, assign importance (learnable weights and biases) to various aspects/objects in the data and be able to differentiate one from the other.	Computer Vision, NLP & Speech processing	Image, video, text, time series	Supervised learning	Classification	Not Explainable	Yes	16, 22, 36, 43, 49, 50, 56, 58, 59; 64, 67, 68, 69, 70, 82, 89, 103, 124, 161

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
DBSCAN	DBSCAN - Density-Based Spatial Clustering of Applications with Noise is a density-based clustering non-parametric algorithm: given a set of points in some space, it groups together points that are closely packed together (points with many nearby neighbours), marking as outliers points that lie alone in low-density regions (whose nearest neighbours are too far away).	Computer Vision	Image	Unsupervised Learning	Clustering			26, 129, 142
Decision tree	A decision tree is a graph that uses a branching method to illustrate every possible output for a specific input in order to break down complex problems.	Classic Data Science	Structured data	Supervised learning	Classification, Regression	Fully Explainable		40, 42, 120,
Deep Q-learning	Deep Q-learning works as Q-learning algorithm at the difference that it uses a neural network to approximate the Q-value function to manage big amount of states and actions.	Classic Data Science	Time series	Reinforcement learning	Rewarding		Yes	65, 85
EfficientNet	EfficientNet is a Convolutional Neural Network based on depth wise convolutions, which makes it lighter than other CNNs. It also allows to scale the model with a unique lever: the compound coefficient.	Computer Vision	Image	Supervised learning	Classification	Not Explainable	Yes	4
Factor analysis of correspondences	The factorial correspondence analysis (CFA) is a statistical method of data analysis which allows the analysis and prioritisation of the information contained in a rectangular table of data and which is today particularly used to study the link between two variables (qualitative or categorical).	Classic Data Science	Structured data	Unsupervised Learning	Dimension Reduction			
GAN	A GAN is a generative model where two networks are placed in competition. The first model is the generator, it generates a sample (e.g. an image), while its opponent, the discriminator, tries to detect whether a sample is real or whether it is the result of the generator. Both improve on the performance of the other.	Computer Vision	Image, Video	Unsupervised Learning				53, 135

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
GMM	A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters.	Computer Vision, NLP & Speech processing	Text, time series, Image, video,	Unsupervised Learning	Clustering			31, 131
GPT-3	Generative Pre-trained Transformer 3 (GPT-3) is an autoregressive language model that uses deep learning to produce human-like text.	NLP & Speech processing	Text	Supervised learning	Classification	Not Explainable	Yes	6
Gradient boosting machine	Gradient boosting is a technique that optimises a decision tree by combining weak models to improve model prediction.	Classic Data Science	Structured data	Supervised learning	Classification, Regression	Globally Explainable		3, 51, 54, 55, 140
Gradient descent	Gradient descent is a first-order iterative optimisation algorithm for finding a local minimum of a differentiable function. The idea is to take repeated steps in the opposite direction of the gradient (or approximate gradient) of the function at the current point, because this is the direction of steepest descent.	Classic Data Science	Structured data	/	Optimisation			17
Graph neural networks (GNNs)	Graph neural networks (GNNs) are deep learning-based methods that operate on graph domain. Graphs are a kind of data structure which models a set of objects (nodes) and their relationships (edges)	Computer Vision, Speech processing	Image	Supervised learning	Regression, classification			20
Hierarchical clustering	Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. The result is a tree-based representation of the objects, named a dendrogram.	Classic Data Science	Structured data	Unsupervised Learning	Clustering			32
Hidden Markov Model (HMM)	Hidden Markov Model is a statistical Markov model in which the system being modelled is assumed to be a Markov process with unobservable hidden states.	Structured data, NLP & Speech processing	Structured data, time series, text	Reinforcement learning	Rewarding		Yes	29
Independent component analysis	ICA is a special case of blind source separation. A common example application is the "cocktail party problem" of listening in on one person's speech in a noisy room.	Classic Data Science	Structured data	Unsupervised Learning	Dimension Reduction			2

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
Isolation forest	The isolation forest returns the anomaly score of each sample. It isolates observations by randomly selecting a feature, and then randomly selecting a split value between the maximum and minimum values of the selected feature.	Classic Data Science	Structured data	Unsupervised learning	Anomaly detection			157, 161
K-means	K-means clustering is a method of vector quantification that aims to partition *n* observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centres or cluster centroid), serving as a prototype of the cluster.	Classic Data Science	Structured data	Unsupervised Learning	Clustering			129
K-Nearest Neighbour	K-Nearest Neighbour is a simple algorithm that stores all the available cases and classifies the new data or case based on a similarity measure. It is mostly used to classify a data point based on how its neighbours are classified.	Classic Data Science	Structured data	Supervised learning	Classification	Fully Explainable	Yes	21, 40,
Linear regression	Linear regression attempts to model the relationship between two or more variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable. For example, a modeller might want to relate the weights of individuals to their heights using a linear regression model.	Classic Data Science	Structured data	Supervised learning	Regression	Fully Explainable		2, 117, 221
Logistic regression	Logistic regression is used to classify data by modelling the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc.	Classic Data Science	Structured data	Supervised learning	Classification	Fully Explainable	Yes	2, 120, 177
LSTM	Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It cannot only process single data points (such as images), but also entire sequences of data (such as speech or video).	NLP & Speech processing, computer vision	Text, image, video	Supervised learning	Regression	Not Explainable		22, 44, 45, 46, 47, 50, 84, 85, 131, 158, 161

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
Mean shift	Mean shift is a non-parametric feature-space analysis technique for locating the maxima of a density function, a so-called mode-seeking algorithm	Computer Vision	Image, video	Unsupervised Learning	Clustering			27
MobileNet	MobileNets are based on a streamlined architecture that uses depth-wise separable convolutions instead of convolutions, in order to build light wFeight deep neural networks.	Computer Vision, Classic Data Science, NLP & Speech processing	Image, video, text, time series, structured data	Unsupervised learning	Clustering		Yes	4
Monte Carlo algorithm	A Monte Carlo algorithm is a randomised algorithm whose output may be incorrect with a certain (typically small) probability.	Classic Data Science	Structured data	Reinforcement learning	Rewarding			30, 105
Multimodal Parallel Network	A Multimodal Parallel Network helps to manage audio-visual event localisation by processing both audio and visual signals at the same time.	Computer Vision, Speech processing	Video	Supervised learning	Classification			18
Naive Bayes classifiers	Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naïve) independence assumptions between the features.	Classic Data Science	Structured data	Supervised learning	Classification	Fully Explainable	Yes	39, 40, 89, 120, 210
Proximal Policy Optimisation	A family of policy gradient methods for Reinforcement Learning that alternate between sampling data and optimising a surrogate objective function using stochastic gradient ascent.	Classic Data Science	Structured data, time series	Reinforcement learning	Rewarding		Yes	137
Principal Component Analysis	The main idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of many variables correlated with each other, either heavily or lightly, while retaining the variation present in the dataset, up to the maximum extent.	Classic Data Science	Structured data	Unsupervised Learning	Dimension Reduction			2
Q-learning	Q-learning is a model-free reinforcement learning algorithm to learn the value of an action in a particular state. It does not require a model of the environment.	Classic Data Science	Structured data, time series	Reinforcement learning	Rewarding		Yes	28
Random forests	Random forests are an ensemble learning method that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.	Classic Data Science	Structured data	Supervised learning	Classification, Regression	Globally Explainable		51, 136, 140

Algorithm Name	Definition	Main domain	Data type	Data environments	Learning Paradigm	Explainability	Accuracy Provided	Refs
Recurrent neural network	A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behaviour.	Computer Vision, NLP & Speech processing	Time series, text, image, video	Supervised learning	Regression	Not Explainable		14, 17, 44, 45, 46, 47, 49, 50, 52, 89, 13
ResNet	A residual neural network (ResNet) is an artificial neural network (ANN) that builds on constructs known from pyramidal cells in the cerebral cortex by utilising skip connections, or shortcuts to jump over some layers.	Computer Vision	Image	Supervised learning	Classification	Not Explainable	Yes	4, 7, 37
Spatial Temporal Graph Convolutional Networks	Spatial Temporal Graph Convolutional Networks is a convolutional neural network that automatically learns both the spatial and temporal patterns from data.	Computer Vision	Video	Supervised learning	Classification			25
Stochastic gradient descent	Stochastic gradient descent is an iterative method for optimising an objective function with suitable smoothness properties. It can be regarded as a stochastic approximation of gradient descent optimisation, since it replaces the actual gradient (calculated from the entire data set) by an estimate thereof (calculated from a randomly selected subset of the data).	Classic Data Science	Structured data	/	Optimisation			17, 24
Support vector machine	SVM are linear classifiers which are based on the margin maximisation principle. They accomplish the classification task by constructing, in a higher dimensional space, the hyperplane that optimally separates data into two categories.	Classic Data Science	Structured data	Supervised learning	Classification	Fully Explainable	Yes	42, 47, 51, 67, 69, 87, 89, 92, 98, 106, 120, 136, 139, 142, 152, 177, 185
WaveNet	Wavenet is a deep neural network for generating raw audio waveforms. The model is fully probabilistic and autoregressive, with the predictive distribution for each audio sample conditioned on all previous ones	NLP & Speech processing	Time series	Unsupervised learning	NLP task			44, 45,131, 132
XGBoost	XGBoost is an extension to gradient boosted decision trees (GBM) and specially designed to improve speed and performance by using regularisation methods to fight overfitting.	Classic Data Science	Structured data	Supervised learning	Classification, Regression	Globally Explainable		3

B ANNEX: MAPPING SECURITY CONTROLS TO THREATS

Threats | *sub-threats*	Vulnerabilities	Security Controls	Threats references
Evasion	Lack of detection of abnormal inputs	Implement tools to detect if a data point is an adversarial example or not	13, 34, 37, 48, 49, 51, 53, 56, 59, 60, 62, 65, 66, 67, 73, 80, 81, 82, 83, 84, 90, 95, 97, 100, 107, 109, 110, 121, 125, 139, 144, 154, 155, 162, 163, 169, 170, 175, 181, 183, 185, 199, 200, 201, 202, 204, 205, 206, 207, 209, 211, 213, 215
		Include ML applications in detection and response to security incident processes	
	Poor consideration of evasion attacks in the model design implementation	Choose and define a more resilient model design	
	Lack of consideration of attacks to which ML applications could be exposed	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	
	Lack of training based on adversarial attacks	Add some adversarial examples to the training dataset	
	Use a widely known model allowing the attacker to study it	Lack of security process to maintain a good security level of the components of the ML application	
		Use less easily transferable models	
		Assess the exposure level of the model used	
	Inputs totally controlled by the attacker which allows for input-output-pairs	Apply modifications to inputs	
Use of adversarial examples crafted in white or grey box conditions (e.g. FGSM…)	Too much information available on the model	Reduce the available information about the model	34, 35, 48, 51, 56, 59, 60, 62, 65, 80, 81, 82, 100, 109, 110, 125, 139, 144, 154, 170, 204, 209
	Too much information about the model given in its outputs	Reduce the information given about the model	
Oracle	Poor access rights management	Apply a RBAC model, respecting the least privileged principle	121, 145, 146, 152, 170, 177, 194, 203, 204, 208, 214
	The model allows private information to be retrieved	Ensure that models respect differential privacy	
	Too much information about the model given in its outputs	Reduce the information given about the model	
	Too much information available on the model	Reduce the available information about the model	
	Lack of consideration of attacks to which ML applications could be exposed to	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	
	Lack of security process to maintain a good security level of the components of the ML application	Implement processes to maintain security levels of ML components over time	
	Weak access protection mechanisms for ML model components	Ensure ML applications comply with identity management, authentication, and process control policies	

Threats | *sub-threats*		Vulnerabilities	Security Controls	Threats references
Poisoning		Model easy to poison	Choose and define a more resilient model design	74, 77, 79, 99, 114, 115, 116, 117, 118, 121, 126, 140, 142, 143, 162, 167, 170, 171, 172, 173, 189, 196, 197, 198, 199, 204, 210
			Implement processes to maintain security levels of ML components over time	
			Assess the exposure level of the model used	
		Lack of data for increasing robustness to poisoning	Enlarge the training dataset	
		Poor access rights management	Apply a RBAC model, respecting the least privileged principle	
		Poor data management	Ensure ML applications comply with data security requirements	
		Undefined indicators of proper functioning, making complex compromise identification	Define and monitor indicators for proper functioning of the model	
		Lack of consideration of attacks to which ML applications could be exposed to	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	
		Use of uncontrolled data	Control all data used by the ML model	
		Use of unsafe data or models (e.g with transfer learning)	Ensure reliable sources are used	
		Lack of control for poisoning	Integrate poisoning control after the "model evaluation" phase	
		No detection of poisoned samples in the training dataset	Use methods to clean the training dataset from suspicious samples	
		Weak access protection mechanisms for ML model components	Ensure ML applications comply with identity management, authentication, and access control policies	
	Label modification	Use of unreliable source to label data	Ensure reliable sources are used	125, 140, 204
Model or data disclosure		Poor access rights management	Apply a RBAC model, respecting the least privileged principle	121, 194, 221, 222
		Existence of unidentified disclosure scenarios	Conduct a risk analysis of the ML application	
		Weak access protection mechanisms for ML model components	Ensure ML applications comply with identity management, authentication, and access control policies	
		Lack of security process to maintain a good security level of the components of the ML application	Implement processes to maintain security levels of ML components over time	
		Unprotected sensitive data on test environments	Ensure appropriate protection are deployed for test environments as well	
	Data disclosure	Too much information about the model given in its outputs	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	
		The model can allow private information to be retrieved	Ensure that models respect differential privacy	
		The model can allow private information to be retrieved	Reduce the information given by the model	
		Disclosure of sensitive data for ML algorithm training	Use federated learning to minimise the risk of data breaches	
	Model disclosure	Too much information available on the model	Reduce the available information about the model	
		Too much information about the model given in its outputs	Reduce the information given by the model	

Threats | *sub-threats*	Vulnerabilities	Security Controls	Threats references
Compromise of ML application components	Poor access rights management	Apply a RBAC model, respecting the least privileged principle	121, 164, 183, 189
	Too much information available on the model	Reduce the available information about the model	
	Existence of several vulnerabilities because the ML application was not integrated into process for integrating security into projects	Ensure ML projects follow the global process for integrating security into projects	
	Use of vulnerable components (among the whole supply chain)	Check the vulnerabilities of the components used so that they have an appropriate security level	
	Too much information about the model given in its outputs	Reduce the information given by the model	
	Existence of unidentified compromise scenarios	Conduct a risk analysis of the ML application	
	Undefined indicators of proper functioning, making complex compromise identification	Define and monitor indicators for proper functioning of the model	
	Bad practices due to a lack of cybersecurity awareness	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	
	Lack of security process to maintain a good security level of the components of the ML application	Implement processes to maintain security levels of ML components over time	
		Ensure ML applications comply with protection policies and are integrated to security operations processes	
	Weak access protection mechanisms for ML model components	Ensure ML applications comply with identity management, authentication, and access control policies	
	Existence of several vulnerabilities because ML specificities are not integrated to existing policies	Integrate ML specificities to existing security policies	
	Existence of several vulnerabilities because ML application do not comply with security policies	Ensure ML applications comply with security policies	
		Include ML applications into asset management processes	
	Contract with a low security third party	Ensure ML applications comply with third parties' security requirements	
Failure or malfunction of ML application	Existing biases in the ML model or in the data	Ensure that models are unbiased	121, 164, 183, 189, 191
	Lack of consideration of real-life conditions in training the model	Ensure that the model is sufficiently resilient to the environment in which it will operate.	
	ML application not integrated in the cyber-resilience strategy	Integrate ML applications into the overall cyber-resilience strategy	
	Existence of unidentified failure scenarios	Conduct a risk analysis of the ML application	
	Undefined indicators of proper functioning, making complex malfunction identification	Define and monitor indicators for proper functioning of the model	
	Lack of explainability and traceability of decisions taken	Build explainable models	
	Lack of security process to maintain a good security level of the components of the ML application	Implement processes to maintain security levels of ML components over time	

Threats | *sub-threats*		Vulnerabilities	Security Controls	Threats references
		Existence of several vulnerabilities because ML specificities are not integrated to existing policies	Ensure ML projects follow the global process for integrating security into projects	
		Contract with a low security third party	Ensure ML applications comply with third parties' security requirements	
		Application not compliant with applicable regulations	Assess the regulations and laws the ML application must comply with	
	Human error	Poor access rights management	Apply a RBAC model, respecting the least privilege principle	
		Lack of documentation on the ML application	Apply documentation requirements to AI projects	
			Include ML applications into asset management processes	
	Denial of service due to inconsistent data or a sponge example	Use of uncontrolled data	Control all data used by the ML model	
	Cybersecurity incident not reported to incident response teams	Lack of cybersecurity awareness	Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	

C ANNEX: IMPLEMENTING SECURITY CONTROLS

Security controls	Examples for operational implementation	References
Add some adversarial examples to the training dataset	The literature provides the following techniques: - Adversarial Training - Ensemble Adversarial Training - Cascade Adversarial Training - Principled Adversarial Training - Gradient Band Based Adversarial Training	13, 23, 48, 51, 59, 65, 72, 95, 108, 162, 200, 201, 202, 211, 215
Apply a RBAC model, respecting the least privilege principle	The NIST 800-53 and the ISO 27001/2 provides several points: - Manage access permissions and authorisations, incorporating the principles of least privilege and separation of duties - Manage the identity of the users (Couple lifecycle management processes and procurement processes etc.)	ISO 27001/2 NIST 800-53, 162
Apply documentation requirements to Artificial Intelligence projects	The NIST 800-53 and the ISO 27001/2 provides several points: - Define change management processes, integrating the update of the documentation	148 ISO 27001/2 NIST 800-53
Apply modifications on inputs	The literature provides the following techniques: - Data randomisation - Input transformation - Input denoising	64, 65, 108, 208
Assess the exposure level of the model used		
Assess the regulations and laws the ML application must comply with	The NIST 800-53 and the ISO 27001/2 provides several points: - Identify applicable legislation - Meet the requirements of GDPR for personal data	ISO 27001/2 NIST 800-53, 162
Build explainable models	The literature provides the following techniques: - Interpret models with some tools - Use model more explainable like regression instead of Deep Neural Network for supervised learning when it is necessary	164, 225, 226, 227

Security controls	Examples for operational implementation	References
Check the vulnerabilities of the components used so that they have an appropriate security level	The NIST 800-53 and the ISO 27001/2 provides several points: - Manage the exemptions by following it industrially, also including remediation plans - Make an inventory of the infrastructure equipment, the applications (Define, document, improve and review a regular process to make inventory) - Manage the maintenance, the obsolete assets etc. (Define a process and continuously improve it, define a roadmap to replace obsolescent technologies) - Implement a vulnerability management policy (Control regularly its implementation) - Perform and manage vulnerability scans on servers OS, middleware, database and network infrastructure (Perform regularly automatics scans)	ISO 27001/2 NIST 800-53
Choose and define a more resilient model design	For poisoning, the literature provides the following technique: - Bagging or weight Bagging - TRIM algorithm For evasion, the literature provides the following technique: - Randomisation - Stability terms into objective function - Adversarial perturbation-based regulariser - Input gradient regularisation - Defensive distillation - Random feature nullification	35, 65, 108, 110, 112, 113, 114, 143, 196, 206, 213
Conduct a risk analysis of the ML application	The NIST 800-53 and the ISO 27001/2 provides several points: - Coordinate the compliance process with legal and audit functions - Identify legal requirements (e.g. GDPR or NIS for European countries) - Establish a methodology to manage identified risks - Establish a formal methodology to analyse risk - Define and monitor the IT resource availability (Formalise a capacity management plan)	ISO 27001/2 NIST 800-53
Control all data used by the ML model	The literature provides the following techniques: - Data sanitisation - RONI and tRONI technics - Point out important data and put a human in the loop (Human in the loop)	114, 116, 118, 142, 162, 197, 228
Define and monitor indicators for proper functioning of the model	The NIST 800-53 and the ISO 27001/2 provides several points: - Formalise a dashboard, bringing together a series of indicators enabling the state of the information system to be judged in relation to the objectives set. - Take actions in case of deviation from the objective - Manage changes on assets, ensure changes will not impact the production and detect any changes in assets' baseline configuration - Guarantee the integrity of the code at all stage (Perform integrity control etc.)	ISO 27001/2 NIST 800-53

Security controls	Examples for operational implementation	References
Enlarge the training dataset	The literature provides the following technique: - Data augmentation	68, 150
Ensure appropriate protection are deployed for test environments as well	The NIST 800-53 and the ISO 27001/2 provides several points: - Protect data when there are in non-production environment (implement desensitisation measures etc.)	ISO 27001/2 NIST 800-53
Ensure that the model is sufficiently resilient to the environment in which it will operate.		
Ensure ML applications comply with Data Security requirements	The NIST 800-53 and the ISO 27001/2 provides several points: - Apply a methodology for data classification. Review the classification regularly - Implement measures to detect data leakage on the Internet (Antivirus, Data right management solution on all sensitive folders) - Secure sensitive data in transit (Deploy mechanisms to detect bypasses on all networks) - Deploy security solutions on network points to prevent data leaks (DLP etc.)	ISO 27001/2 NIST 800-53
Ensure ML applications comply with identity management, authentication and access control policies	The NIST 800-53 and the ISO 27001/2 provides several points: - Define a policy regarding users' authentication (Define an authentication policy that considers the sensitivity of resources and the connection context for all types of account, use Multi-Factor Authentication) - Define a remote access policy (Verify security configuration, authenticate connected devices)	ISO 27001/2 NIST 800-53, 162
Ensure ML applications comply with protection policies and are integrated to security operations processes	The NIST 800-53 and the ISO 27001/2 provides several points: - Manage the maintenance, the obsolete assets etc. (Define a process and continuously improve it, define a roadmap to replace obsolescent technologies) - Implement a vulnerability management policy (Control regularly its implementation) - Perform and manage vulnerability scans on servers OS, middleware, database and network infrastructure (Perform regularly automatics scans)	ISO 27001/2 NIST 800-53
Ensure ML applications comply with security policies	The NIST 800-53 and the ISO 27001/2 provides the following point: - Define policies for information security	ISO 27001/2 NIST 800-53
Ensure ML applications comply with third parties' security requirements	The NIST 800-53 and the ISO 27001/2 provides several points: - Integrate the security into contracts (Define a security insurance plan for strategic contracts representing high risks for company security. Communicate roles and responsibilities to every new third party) - Monitor and review third parties services	ISO 27001/2 NIST 800-53

Security controls	Examples for operational implementation	References
Ensure ML projects follow the global process for integrating security into projects	The NIST 800-53 and the ISO 27001/2 provides several points: - Integrate the security into contracts (Define a security insurance plan for strategic contracts representing high risks for company security. Communicate roles and responsibilities to every new third party) - Define and manage a patch management policy - Manage the interconnections with external systems (establish a formal process to regularly review the exhaustiveness of interconnections inventory) - Integrate and manage security protection for applications (firewalls, WAF, reverse proxy) - Perform security controls on application	ISO 27001/2 NIST 800-53
Ensure reliable sources are used	The NIST 800-53 and the ISO 27001/2 provides several points: - Manage the interconnections with external systems (establish a formal process to regularly review the exhaustiveness of interconnections inventory)	ISO 27001/2 NIST 800-53
Ensure that models are unbiased	The literature provides the following techniques: - Classification parity - Calibration - Anti-classification - Having a diverse dataset - Some other technics: samples bias, measurement error…	217, 229
Ensure that models respect differential privacy to a sufficient degree	The literature provides the following techniques: - Model design adapted like PATE for Deep Neural Network based classifier - Data randomisation - Randomisation - Objective function perturbation	63, 108, 112, 194, 203, 208, 214
Implement processes to maintain security levels of ML components over time	The NIST 800-53 and the ISO 27001/2 provides the following point: - Perform technical and organisational audits regularly on critical scope and develop an action plan after each audit	ISO 27001/2 NIST 800-53
Implement tools to detect if a data point is an adversarial example or not	The literature provides the following technique: - Adding detector subnetworks	65, 207
Include ML applications into asset management processes	The NIST 800-53 and the ISO 27001/2 provides several points: - Create an inventory of the infrastructure equipment, the applications (Define, document, improve and review a regular process to make inventory) - Classify information - Manage the maintenance, the obsolete assets etc. (Define a process and continuously improve it, define a roadmap to replace obsolescent technologies)	ISO 27001/2 NIST 800-53

Security controls	Examples for operational implementation	References
Include ML applications into detection and response to security incident processes	The NIST 800-53 and the ISO 27001/2 provides several points: - Include ML projects into the Business Continuity Plan - Include ML projects into the Cybersecurity Disaster Recovery Plan - Define a backup strategy for ML projects (and test it) - Define a strategy for public relations during recovery (Identify and train possible spokespersons, Adapt communication responses to different categories of interlocutors)	ISO 27001/2 NIST 800-53
Integrate ML applications into the overall cyber-resilience strategy	The NIST 800-53 and the ISO 27001/2 provides several points: - Include ML projects into the Business Continuity Plan - Include ML projects into the Cybersecurity Disaster Recovery Plan - Define a backup strategy for ML projects (and test it) - Define a strategy for public relations during recovery (Identify and train possible spokespersons, Adapt communication responses to different categories of interlocutors)	ISO 27001/2 NIST 800-53, 162
Integrate ML specificities to existing security policies	The NIST 800-53 and the ISO 27001/2 provides the following point: - Review policies for information security	ISO 27001/2 NIST 800-53
Integrate ML specificities to awareness strategy and ensure all ML stakeholders are receiving it	The NIST 800-53 and the ISO 27001/2 provides several points: - Organise training sessions - Perform locally cyber risks reporting	ISO 27001/2 NIST 800-53
Integrate poisoning control after the "model evaluation" phase	The literature provides the following technique: - STRIP technique	198
Reduce the available information about the model	The NIST 800-53 and the ISO 27001/2 provides the following point: - Implement a classification policy	ISO 27001/2 NIST 800-53
Reduce the information given by the model	The literature provides the following technique: - Gradient Masking	89, 145
Use federated learning to minimise risk of data breaches		194
Use less easily transferable models		65, 215
Use methods to clean the training dataset from suspicious samples	The literature provides the following techniques: - Data sanitisation - RONI and tRONI technics - Point out important data and put a human in the loop (Human in the loop)	114, 162, 210

D ANNEX: REFERENCES

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
1	Adversarial Machine Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning	https://www.morganclaypool.com/doi/abs/10.2200/S00861ED1V01Y201806AIM039	2018	• Yevgeniy Vorobeychik • Murat Kantarcioglu						X				
2	The Elements of Statistical Learning	https://web.stanford.edu/~hastie/Papers/ESLII.pdf	2001	• Trevor Hastie • Robert Tibshirani • Jerome Friedman	X	X	X	X	X	X	X	X	X	
3	XGBoost: a scalable tree boosting system	https://arxiv.org/pdf/1603.02754.pdf	2016	• Tianqi Chen • Carlos Guestrin					X		X			
4	EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks	https://proceedings.mlr.press/v97/tan19a/tan19a.pdf	2019	• Mingxing Tan • Quoc V. Le		X					X			
5	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	https://arxiv.org/pdf/1810.04805.pdf	2019	• Jacob Devlin • Ming-Wei Chang • Kenton Lee • Kristina Toutanova			X					X		
6	Language Models are Few-Shot Learners	https://arxiv.org/pdf/2005.14165.pdf	2020	• Tom B. Brown • Benjamin Mann • Nick Ryder • Melanie Subbiah • Jared Kaplan • Prafulla Dhariwal • Arvind Neelakantan • Pranav Shyam • Girish Sastry • Amanda Askell • Sandhini Agarwal • Ariel Herbert-Voss • Gretchen Krueger • Tom Henighan • Rewon Child • Aditya Ramesh • Daniel M. Ziegler • Jeffrey Wu • Clemens Winter • Christopher Hesse • Mark Chen • Eric Sigler • Mateusz Litwin • Scott Gray • Benjamin Chess • Jack Clark • Christopher			X					X		

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
				Berner • Sam McCandlish • Alec Radford • Ilya Sutskeve • Dario Amodei										
7	Deep Double Descent: Where Bigger Models and More Data Hurt	https://arxiv.org/pdf/1912.02292.pdf	2019	• Preetum Nakkiran • Gal Kaplun • Yamini Bansal • Tristan Yang • Boaz Barak • Ilya Sutskever		X					X			
8	Deep Residual Learning for Image Recognition	https://arxiv.org/pdf/1512.03385.pdf	2015	• Kaiming He • Xiangyu Zhang • Shaoqing Re • Jian Sun		X					X			
9	Practical Deep Learning with Bayesian Principles	https://papers.nips.cc/paper/2019/file/b53477c2821c1bf0da5d40e57b870d35-Paper.pdf	2019	• Kazuki Osawa • Siddharth Swaroop • Anirudh Jain • Runa Eschenhagen • Richard E. Turner • Rio Yokota • Mohammad Emtiyaz Khan		X					X			
10	Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps	https://arxiv.org/pdf/1312.6034.pdf	2014	• Karen Simonyan • Andrea Vedaldi • Andrew Zisserman		X					X			
11	Reinforcement learning: An introduction	https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf	1998	• Richard S. Sutton • Andrew G. Barto	X	X			X					X
12	Exploration by Random Network Distillation	https://arxiv.org/pdf/1810.12894.pdf	2018	• Yuri Burda • Harrison Edwards • Amos Storkey • Oleg Klimov		X								X
13	DeepFool: a simple and accurate method to fool deep neural networks, in arXiv, July 2016	https://arxiv.org/pdf/1511.04599.pdf	2016	• Seyed-Mohsen • Moosavi-Dezfooli • Alhussein Fawzi • Pascal Frossard							X			
14	Dual Learning for Machine Translation	https://arxiv.org/pdf/1611.00179.pdf	2016	• Yingce Xia • Di He • Tao Qin • Liwei Wang • Nenghai Yu • Tie-Yan Liu • Wei-Ying Ma			X							X

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
15	Video Prediction via Example Guidance	https://arxiv.org/pdf/2007.01738.pdf	2020	• Jingwei Xu • Huazhe Xu • Bingbing Ni • Xiaokang Yang • Trevor Darrell	X					X				
16	Context-aware Attentional Pooling (CAP) for Fine-grained Visual Classification	https://arxiv.org/pdf/2101.06635.pdf	2021	• Ardhendu Behera • Zachary Wharton • Pradeep Hewage • Asish Bera		X					X			
17	On the importance of initialisation and momentum in deep learning	http://proceedings.mlr.press/v28/sutskever13.pdf	2013	Ilya Sutskever/James Martens /George Dahl/Geo↵rey Hinton	X	X	X	X	X					
18	MPN: MULTIMODAL PARALLEL NETWORK FOR AUDIO-VISUAL EVENT LOCALISATION	https://arxiv.org/pdf/2104.02971.pdf	2021	• Jiashuo Yu • Ying Cheng • Rui Feng	X			X			X			
19	Zero-Gradient Constrained Optimisation for Destriping of 3D Imaging Data	https://arxiv.org/pdf/2104.02845.pdf	2021	• Kazuki Naganuma • Shunsuke Ono	X	X								
20	Attentional Graph Neural Network for Parking-slot Detection	https://arxiv.org/pdf/2104.02576.pdf	2021	• Chen Min • Jiaolong Xu • Liang Xiao • Dawei Zhao • Yiming Nie • Bin Dai		X					X			
21	Identity and Posture Recognition in Smart Beds with Deep Multitask Learning	https://arxiv.org/pdf/2104.02159.pdf	2019	• Vandad Davoodnia • Ali Etemad		X					X			
22	A Combined CNN and LSTM Model for Arabic Sentiment Analysis	https://arxiv.org/pdf/1807.02911.pdf	2018	• Abdulaziz M. Alayba • Vasile Palade • Matthew England • Rahat Iqbal			X				X			
23	Adversarial Training is Not Ready for Robot Learning	https://arxiv.org/abs/2103.08187	2021	• Mathias Lechner • Ramin Hasani • Radu Grosu • Daniela Rus • Thomas A. Henzinger	X	X				X				
24	Improved Adam Optimizer for Deep Neural Networks	http://iwqos2018.ieee-iwqos.org/files/2018/05/Improved_Adam_Optimizer.pdf	2018	• Zijun Zhang	X	X	X	X	X					
25	Vision-Based Fall Detection Using ST-GCN	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9351913	2021	• Oussema Keskes • Rita Noumeir	X						X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
26	DBSCAN++: Towards fast and scalable density clustering	https://arxiv.org/pdf/1810.13105.pdf	2019	• Jennifer Jang • Heinrich Jiang	X							X		
27	MeanShift++: Extremely Fast Mode-Seeking With Applications to Segmentation and Object Tracking	https://arxiv.org/pdf/2104.00303.pdf	2021	• Jennifer Jang • Heinrich Jiang	X	X						X		
28	Self-correcting Q-Learning	https://arxiv.org/pdf/2012.01100.pdf	2021	• Rong Zhu • Mattia Rigotti					X					X
29	A New Algorithm for Hidden Markov Models Learning Problem	https://arxiv.org/ftp/arxiv/papers/2102/2102.07112.pdf	2021	• Taha Mansouri • Mohamadreza Sadeghimoghadam • Iman Ghasemian Sahebi			X	X	X					X
30	Deep Reinforcement Learning Aided Monte Carlo Tree Search for MIMO Detection	https://arxiv.org/pdf/2102.00178.pdf	2021	• Tz-Wei Mo • Ronald Y. Chang • Te-Yi Kan				X	X					X
31	Hard-Clustering with Gaussian Mixture Models	https://arxiv.org/pdf/1603.06478.pdf	2016	• Johannes Blomer • Sascha Brauer • Kathrin Bujna	X	X	X	X	X			X		
32	Analysis of Agglomerative Clustering	https://arxiv.org/pdf/1012.3697.pdf	2014	• Marcel R. Ackermann • Johannes Blomer • Daniel Kuntze • Christian Sohler					X			X		
33	Towards Evaluating the Robustness of Neural Networks	https://arxiv.org/pdf/1608.04644.pdf	2017	• Nicholas Carlini • David Wagner		X					X			
34	DELVING INTO TRANSFERABLE ADVERSARIAL EXAMPLES AND BLACK-BOX ATTACKS	https://arxiv.org/pdf/1611.02770.pdf	2017	• Yanpei Liu • Xinyun Chen • Chang Liu • Dawn Song		X					X			
35	Generating Adversarial Examples with Adversarial Networks	https://arxiv.org/pdf/1801.02610.pdf	2019	• Chaowei Xiao • Bo Li • Jun-Yan Zhu • Warren He • Mingyan Liu • Dawn Song		X					X			
36	Detecting Overfitting via Adversarial Examples	https://arxiv.org/pdf/1903.02380.pdf	2019	• Roman Werpachowski • András György • Csaba Szepesvári		X					X			
37	Cybersecurity Challenges in the Uptake of Artificial Intelligence in Autonomous Driving	https://www.enisa.europa.eu/publications/enisa-jrc-cybersecurity-challenges-in-the-	2021	• Georgia Dede • Ronan Hamon • Rossen Naydenov • Henrik Junklewitz		X					X			X

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
		uptake-of-artificial-intelligence-in-autonomous-driving		• Apostolos Malatras • Ignacio Sanchez										
38	Rapid Object Detection using a Boosted Cascade of Simple Features	https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf	2001	• Paul Viola • Michael Jones		X					X			
39	Detection of Advanced Malware by Machine Learning Techniques	https://arxiv.org/ftp/arxiv/papers/1903/1903.02966.pdf	2019	• Sanjay Sharma • C. Rama Krishna • Sanjay K. Sahay			X		X		X			
40	MACHINE LEARNING METHODS FOR MALWARE DETECTION AND CLASSIFICATION	https://www.theseus.fi/bitstream/handle/10024/123412/Thesis_final.pdf?sequence=1&isAllowed=y	2017	• Kateryna Chumachenko			X		X		X			
41	Adversarial Malware Binaries: Evading Deep Learning for Malware Detection in Executables	https://arxiv.org/pdf/1803.04173.pdf	2018	• Bojan Kolosnjaji • Ambra Demontis • Battista Biggio • Davide Maiorca • Giorgio Giacinto • Claudia Eckert • Fabio Roli			X		X		X			
42	Generating Adversarial Malware Examples for Black-Box Attacks Based on GAN	https://arxiv.org/pdf/1702.05983.pdf	2017	• Weiwei Hu • Ying Tan			X		X		X			
43	Exploring Adversarial Examples in Malware Detection	https://arxiv.org/pdf/1810.08280.pdf	2018	• Octavian Suciu • Scott E. Coull • Jeffrey Johns			X		X		X			
44	Deceiving End-to-End Deep Learning Malware Detectors using Adversarial Examples	https://arxiv.org/pdf/1802.04528.pdf	2019	• Felix Kreuk • Assi Barak • Shir Aviv-Reuven • Moran Baruch • Benny Pinkas • Joseph Keshet			X		X		X			
45	Malware Detection by Eating a Whole EXE	https://arxiv.org/pdf/1710.09435.pdf	2017	•Edward Raff •Jon Barker •Jared Sylvester •Robert Brandon •Bryan Catanzaro •Charles Nicholas			X		X		X			
46	Black-box attacks against rnn based malware detection algorithms	https://arxiv.org/pdf/1705.08131.pdf	2017	•Weiwei Hu •Ying Tan			X		X		X			
47	Generic black-box end-to-end attack against rnns and other API calls based malware classifiers	https://arxiv.org/pdf/1707.05970.pdf	2018	• Shai Rosenberg • Asaf Shabtai • Lior Rokach •Yuval Elovici			X		X		X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
48	Universal Adversarial Perturbations for Malware	https://arxiv.org/pdf/2102.06747.pdf	2021	• Raphael Labaca-Castro • Luis Muñoz-González •Feargus Pendlebury •Gabi Dreo Rodosek •Fabio Pierazzi • Lorenzo Cavallaro			X		X		X			
49	MDEA: Malware Detection with Evolutionary Adversarial Learning	https://arxiv.org/pdf/2002.03331.pdf	2020	• Xiruo Wang • Risto Miikkulainen			X		X					X
50	Attack and Defense of Dynamic Analysis-Based, Adversarial Neural Malware Classification Models	https://arxiv.org/pdf/1712.05919.pdf	2017	• Jack W. Stokes • De Wang, Mady Marinescu • Marc Marino • Brian Bussone			X		X		X			
51	Robust Android Malware Detection System against Adversarial Attacks using Q-Learning	https://arxiv.org/pdf/2101.12031.pdf	2021	• Hemant Rathore • Sanjay K. Sahay • Piyush Nikam • Mohit Sewak			X		X		X			
52	Binary Black-box Evasion Attacks Against Deep Learning-based Static Malware Detectors with Adversarial Byte-Level Language Model	https://arxiv.org/pdf/2012.07994.pdf	2020	• Mohammadreza Ebrahimi • Ning Zhang, James Hu • Muhammad Taqi Raza • Hsinchun Chen			X		X		X			
53	MalFox: Camouflaged Adversarial Malware Example Generation Based on C-GANs Against Black-Box Detectors	https://arxiv.org/pdf/2011.01509.pdf	2020	•Fangtian Zhong • Xiuzhen Cheng • Dongxiao YuBei Gong • Shuaiwen Song • Jiguo Yu			X		X		X			
54	Generating End-to-End Adversarial Examples for Malware Classifiers Using Explainability	https://arxiv.org/pdf/2009.13243.pdf	2020	• Ishai Rosenberg • Shai Meir, Jonathan Berrebi • Ilay Gordon • Guillaume Sicard • Eli (Omid)David			X		X		X			
55	Adversarial EXEmples: A Survey and Experimental Evaluation of Practical Attacks on Machine Learning for Windows Malware Detection	https://arxiv.org/pdf/2008.07125.pdf	2020	• Luca Demetrio • Scott E. Coull • Battista Biggio • Giovanni Lagorio • Alessandro Armando • Fabio Roli			X		X		X			
56	Adversarial Machine Learning Attacks and Defense Methods in the Cyber Security Domain	https://arxiv.org/pdf/2007.02407.pdf	2020	• Ihai Rosenberg • Asaf Shabtai • Yuval Elovici • Lior Rokach			X		X		X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
57	Automatic Generation of Adversarial Examples for Interpreting Malware Classifiers	https://arxiv.org/pdf/2003.03100.pdf	2020	• Wei Song • Xuezixiang Li • Sadia Afroz • Deepali Garg • Dmitry Kuznetsov • Heng Yin			X		X		X			
58	COPYCAT: Practical Adversarial Attacks on Visualisation-Based Malware Detection	https://arxiv.org/pdf/1909.09735.pdf	2019	• Aminollah Khormali • Ahmed Abusnaina • Songqing Chen • DaeHun Nyang • Aziz Mohaisen			X		X		X			
59	Effectiveness of Adversarial Examples and Defenses for Malware Classification	https://arxiv.org/pdf/1909.04778.pdf	2019	• Robert Podschwadt • Hassan Takabi			X		X		X			
60	Generating Adversarial Computer Programs using Optimiser Obfuscations	https://arxiv.org/pdf/2103.11882.pdf	2021	• Shashank Srikant • Sijia Liu • Tamara Mitrovska • Shiyu Chang • Quanfu Fan • Gaoyuan Zhang • Una-May O'Reilly			X		X		X			
61	Adversarial Robustness with Non-uniform Perturbations	https://arxiv.org/pdf/2102.12002.pdf	2021	• Ecenaz Erdemir • Jeffrey Bickford • Luca Melis • Sergul Aydore			X		X		X			
62	Robust Neural Networks using Randomiser Adversarial Training	https://hal.archives-ouvertes.fr/hal-02380184v2/document	2019	• Alexandre Araujo • Laurent Meunier • Rafael Pinot • Benjamin Negrevergne		X					X			
63	Theoretical evidence for adversarial robustness through randomisation	https://arxiv.org/pdf/1902.01148.pdf	2019	•Rafael Pinot • Laurent Meunier • Alexandre Araujo • Hisashi Kashima • Florian Yger • Cédric Gouy-Pailler • Jamal Atif		X					X			
64	Mitigating Adversarial Effects Through Randomisation	https://arxiv.org/pdf/1711.01991.pdf	2017	• Cihang Xie • Jianyu Wang • Zhishuai Zhang • Zhou Ren • Alan Yuille		X					X			
65	Adversarial attack and defense in reinforcement learning-from AI security view	https://cybersecurity.springeropen.com/track/pdf/10.1186/s42400-019-0027-x.pdf	2019	•Tong Chen • Jiqiang Liu • Yingxiao Xiang • Wenjia Niu • Endong Tong and Zhen Han	X	X		X	X					X

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
66	TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP	https://arxiv.org/pdf/2005.05909.pdf	2020	• John X. Morris • Eli Lifland • Jin Yong Yoo • Jake Grigsby • Di Jin • Yanjun Qi			X							X
67	Adversarial Attacks and Defences: A Survey	https://arxiv.org/pdf/1810.00069.pdf	2018	• Anirban Chakraborty • Manaar Alam • Vishal Dey • Anupam Chattopadhyay • Debdeep Mukhopadhyay		X					X			
68	DATA AUGMENTATION BASED MALWARE DETECTION USING CONVOLUTIONAL NEURAL NETWORKS	https://arxiv.org/pdf/2010.01862.pdf	2021	• Ferhat Ozgur Catak • Javed Ahmed • Kevser Sahinbas • Zahid Hussain Khand		X	X				X			
69	A Taxonomy and Survey of Attacks Against Machine Learning	https://gala.gre.ac.uk/id/eprint/25226/7/25226%20LOUKAS_Taxonomy_And_Survey_Of_Attacks_Against_Machine_Learning_%28AAM%29_2019.pdf	2019	• Nikolaos Pitropakisa • Emmanouil Panaousisb • Thanassis Giannetsosc • Eleftherios Anastasiadisd • George Loukase		X					X			
70	Adversarial examples are not bugs, they are features.	https://arxiv.org/pdf/1905.02175.pdf	2019	• Andrew Ilyas • Shibani Santurkar • Dimitris Tsipras • Logan Engstrom • Brandon Tran • Aleksander Madry		X					X			
71	Impact of Spatial Frequency Based Constraints on Adversarial Robustness	https://arxiv.org/pdf/2104.12679.pdf	2021	• Rémi Bernhard • Pierre-Alain Moellic • Martial Mermillod • Yannick Bourrier • Romain Cohendet • Miguel Solinas • Marina Reyboz		X					X			
72	Improving Adversarial Robustness Using Proxy Distributions	https://arxiv.org/pdf/2104.09425.pdf	2021	• Vikash Sehwag • Saeed Mahloujifar • Tinashe Handina • Sihui Dai • Chong Xiang • Mung Chiang • Prateek Mittal		X					X			
73	Improving Adversarial Transferability with Gradient Refining	https://arxiv.org/pdf/2105.04834.pdf	2021	• Guoqiu Wang • Huanqian Yan • Ying Guo • Xingxing Wei		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
74	Poisoning MorphNet for Clean-Label Backdoor Attack to Point Clouds	https://arxiv.org/pdf/2105.04839.pdf	2021	• Guiyu Tian • Wenhao Jiang • Wei Liu • Yadong Mu		X					X			
75	High-Robustness, Low-Transferability Fingerprinting of Neural Networks	https://arxiv.org/pdf/2105.07078.pdf	2021	• Siyue Wang • Xiao Wang • Pin-Yu Chen • Pu Zhao • Xue Lin		X					X			
76	Detecting Adversarial Examples with Bayesian Neural Network	https://arxiv.org/pdf/2105.08620.pdf	2021	• Yao Li • Tongyi Tang • Cho-Jui Hsieh • Thomas C. M. Lee		X					X			
77	Robust Backdoor Attacks against Deep Neural Networks in Real Physical World	https://arxiv.org/pdf/2104.07395.pdf	2021	• Mingfu Xue • Can He • Shichang Sun • Jian Wang • Weiqiang Liu		X					X			
78	secml-malware: A Python Library for Adversarial Robustness Evaluation of Windows Malware Classifiers	https://arxiv.org/pdf/2104.12848.pdf	2021	• Luca Demetrio • Battista Biggio			X		X		X			
79	Defending against adversarial denial-of-service data poisoning attack	https://arxiv.org/pdf/2104.06744.pdf	2021	• Nicolas M. Müller • Simon Roschmann • Konstantin Böttinger		X					X			
80	Attack on practical speaker verification system using universal adversarial perturbations	https://arxiv.org/pdf/2105.09022.pdf	2021	• Weiyi Zhang • Shuning Zhao • Le Liu • Jianmin Li • Xingliang Cheng • Thomas Fang Zheng • Xiaolin Hu					X		X			
81	Exploiting Vulnerabilities in Deep Neural Networks: Adversarial and Fault-Injection Attacks	https://arxiv.org/pdf/2105.03251.pdf	2021	• Faiq Khalid • Muhammad Abdullah Hanif • Muhammad Shafique		X					X			
82	Dynamic Defense Approach for Adversarial Robustness in Deep Neural Networks via Stochastic Ensemble Smoothed Model	https://arxiv.org/ftp/arxiv/papers/2105/2105.02803.pdf	2021	• Ruoxi Qin • Linyuan Wang • Xingyuan Chen • Xuehui Du • Bin Yan		X					X			
83	Adv-Makeup: A New Imperceptible and Transferable Attack on Face Recognition	https://arxiv.org/pdf/2105.03162.pdf	2021	• Bangjie Yin • Wenxuan Wang • Taiping Yao • Junfeng Guo • Zelun Kong • Shouhong Ding • Jilin Li • Cong Liu		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
84	Adversarial Attacks on Machine Learning Systems for High-Frequency Trading	https://arxiv.org/pdf/2002.09565.pdf	2020	• Micah Goldblum • Avi Schwarzschild • Ankit B. Patel • Tom Goldstein			X		X		X			
85	M. Hausknecht & Al., Deep Recurrent Q-Learning for Partially Observable MDPs, in arXiv, January 2017	https://arxiv.org/pdf/1507.06527.pdf	2017	• Matthew Hausknecht • Peter Stone	X	X								X
86	Learning Malware Models via Reinforcement Learnin	https://arxiv.org/pdf/1801.08917.pdf	2018	• Hyrum S. Anderson • Hyrum S. Anderson • Bobby Filar • David Evans • Phil Roth			X		X					X
87	Wild patterns: ten years after the rise of adversarial Machine Learning	https://arxiv.org/pdf/1712.03141.pdf	2018	• Battista Biggioa • Fabio Rolia		X					X			
88	A survey on security threats and defensive techniques of machine learning: a data driven view	https://www.researchgate.net/publication/323154427_A_Survey_on_Security_Threats_and_Defensive_Techniques_of_Machine_Learning_A_Data_Driven_View	2018	• Qiang Liu • Pan Li • Wentao Zhao • Wei Cai • Shui Yu • Victor C. M. Leung	X	X	X	X	X	X	X	X	X	
89	Adversarial attacks and defenses in images, graphs and text: a review	https://arxiv.org/pdf/1909.08072.pdf	2020	• Han Xu • Yao Ma • Haochen Liu • Debayan Deb • Hui Liu • Jiliang Tang • Anil K. Jain		X	X			X	X	x	X	X
90	Adversarial Learning	https://www.researchgate.net/publication/221654486_Adversarial_learning	2005	• Daniel Lowd • Christopher Meek					X		X			
91	Adversarial Machine-Learning	https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.360.168&rep=rep1&type=pdf	2011	• Ling Huang • Anthony D. Joseph • Blaine Nelson • Benjamin I. P. Rubinstein • J. D. Tygar							X	X		
92	A Survey of Adversarial Machine Learning in Cyber Warfare	https://www.researchgate.net/publication/327074374_A_Survey_of_Adversarial_Machine_Learning_in_Cyber_Warfare	2018	• Vasisht Duddu	X	X	X	X	X	X	X	X	X	X

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
93	Intelligence artificielle et cybersécurité : protéger dès maintenant le monde de demain	https://www.wavestone.com/app/uploads/2019/09/IA-cyber-2019.pdf	2019	• Carole Meziat • Laurent Guille						X	X	X	X	X
94	Explaining and Harnessing Adversarial Examples	https://arxiv.org/pdf/1412.6572.pdf	2020	• Ian J. Goodfellow • Jonathon Shlens • Christian Szegedy		X					X			
95	Feature Cross-Substitution in Adversarial Classification	https://papers.nips.cc/paper/2014/file/8597a6cfa74defcbde3047c891d78f90-Paper.pdf	2014	• Yevgeniy Vorobeychik • Bo Li			X				X			
96	Adversarial Examples in Deep Learning: Characterisation and Divergence	http://arxiv.org/abs/1807.00051	2018	• Wenqi Wei • Ling Liu • Margaret Loper • Stacey Truex • Lei Yu • Mehmet Emre Gursoy • Yanzhao Wu		X					X			
97	Maximal Jacobian-based Saliency Map Attack	http://arxiv.org/abs/1808.07945	2018	• Rey Wiyatno • Anqi Xu		X					X			
98	Robustness and Regularisation of Support Vector Machines	https://www.jmlr.org/papers/volume10/xu09b/xu09b.pdf	2009	• Huan Xu • Constantine Caramanis • Shie Mannor										
99	Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning	https://arxiv.org/pdf/1712.05526.pdf	2017	• Xinyun Chen • Chang Liu • Bo Li • Kimberly Lu • Dawn Song		X					X			
100	Towards evaluating the robustness of neural networks	https://arxiv.org/abs/1608.04644	2016	• Nicholas Carlini • David Wagner		X					X			
101	Detecting adversarial samples from artifacts	https://arxiv.org/abs/1703.00410	2017	• Reuben Feinman • Ryan R. Curtin • Saurabh Shintre • Andrew B. Gardner		X					X			
102	Single Headed Attention RNN: Stop Thinking With Your Head	https://arxiv.org/pdf/1911.11423.pdf	2019	• Steven Merity			X				X			
103	Semantic Segmentation using Adversarial Networks	https://arxiv.org/pdf/1611.08408.pdf	2016	• Pauline Luc • Camille Couprie • Soumith Chintala • Jakob Verbeek		X					X			
104	Unorganiser Malicious Attacks Detection	https://arxiv.org/pdf/1610.04086.pdf	2018	• Ming Pang • Wei Gao • Min Tao • Zhi-Hua Zhou										

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
105	An Introduction to MCMC for Machine Learning	https://link.springer.com/content/pdf/10.1023/A:1020281327116.pdf	2003	• CHRISTOPHE ANDRIEU • NANDO DE FREITAS • ARNAUD DOUCET • MICHAEL I. JORDAN										
106	Convex Learning with Invariances	http://people.csail.mit.edu/gamir/pubs/TeoGloRowSmo07.pdf	2007	• Choon Hui Teo • Amir Globerson • Sam Roweis • Alexander J. Smola		X					X			
107	Scalable Optimisation of Randomiser Operational Decisions in Adversarial Classification Settings	http://proceedings.mlr.press/v38/li15a.pdf	2015	• Bo Li • Yevgeniy Vorobeychik					X		x			
108	Adversarial Attacks and Defenses in Deep Learning,	https://www.sciencedirect.com/science/article/pii/S2095809919305030X	2020	• Kui Ren • Tianhang Zheng • Zhan Qin • Xue Liu	X	X	X	X			X			
109	Exploring the Space of Adversarial Images	https://arxiv.org/pdf/1510.05328.pdf	2016	• Pedro Tabacof • Eduardo Valle		X					X			
110	Distillation as a Defense to Adversarial Perturbations against Deep Neural Network	https://arxiv.org/pdf/1511.04508.pdf	2016	• Nicolas Papernot • Patrick McDaniel • Xi Wu • Somesh Jha • Ananthram Swami		X					X			
111	Characterizing adversarial subspaces using local intrinsic dimensionality	https://arxiv.org/pdf/1801.02613.pdf	2018	• Xingjun Ma • Bo Li • Yisen Wang • Sarah M. Erfani • Sudanthi Wijewickrema • Grant Schoenebeck • Dawn Song • Michael E. Houle • James Bailey		X					X			
112	Defense still has a long way: Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods	https://arxiv.org/pdf/1705.07263.pdf	2017	• Nicholas Carlini • David Wagner		X					X			
113	Towards Deep Learning Models Resistant to Adversarial Attacks	https://arxiv.org/pdf/1706.06083.pdf	2019	• Aleksander Madry • Aleksandar Makelov • Ludwig Schmidt • Dimitris Tsipras • Adrian Vladu		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
114	Poisoning attacks on Machine Learning	https://towardsdatascience.com/poisoning-attacks-on-Machine-Learning-1ff247c254db	2019	• Ilja Moisejevs							X			
115	A Unified Framework for Data Poisoning Attack to Graph-based Semi-supervised Learning	https://arxiv.org/pdf/1910.14147.pdf	2019	• Xuanqing Liu • Si Si • Xiaojin Zhu • Yang Li • Cho-Jui Hsieh		X				X	X			
116	Data Poisoning Attacks on Multi-Task Relationship Learning	https://personal.ntu.edu.sg/boan/papers/AAAI18_MTL.pdf	2018	• Mengchen Zhao • Bo An • Yaodong Yu • Shulin Liu • Sinno Jialin Pan		X				X	X			
117	Robust High-Dimensional Linear Regression	https://arxiv.org/pdf/1608.02257.pdf	2016	• Chang Liu • Bo Li • Yevgeniy Vorobeychik • Alina Oprea					X	X				
118	Certified Defenses for Data Poisoning Attacks	https://arxiv.org/pdf/1706.03691.pdf	2017	• Jacob Steinhardt • Pang Wei Koh • Percy Liang		X	X				X			
119	Generative adversarial nets	https://arxiv.org/pdf/1406.2661.pdf	2014	• Ian J. Goodfellow • Jean Pouget-Abadie • Mehdi Mirza • Bing Xu • David Warde-Farley • Sherjil Ozair • Aaron Courville • Yoshua Bengio		X					X			
120	A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8290925	2018	• Qiang Liu • Pan Li • Wentao Zhao • Wei Cai • Shui Yu • Vctor C. M. Leung		X				X	X	X		
121	Artificial Intelligence Cybersecurity Challenges	https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges	2021	• Apostolos Malatras • Georgia Dede										
122	A Taxonomy of ML for Systems Problems	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9153088	2020	• Martin Maas							X	X	X	X

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
123	Deep Unsupervised Learning for Generaliser Assignment Problems: A Case-Study of User-Association in Wireless Networks	https://arxiv.org/pdf/2103.14548.pdf	2021	• Arjun Kaushik • Mehrazin Alizadeh • Omer Waqar • Hina Tabassum					X			X	X	
124	Attacks against machine learning — an overview	https://elie.net/blog/ai/attacks-against-machine-learning-an-over	2018	• Elie Bursztein							X			
125	How to attack Machine Learning (Evasion, Poisoning, Inference, Trojans, Backdoors)	https://towardsdatascience.com/how-to-attack-machine-learning-evasion-poisoning-inference-trojans-backdoors-a7cb5832595c	2019	• Alex Polyakov		X					X			
126	Trojaning Attack on Neural Networks	https://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=2782&context=cstech	2017	• Yingqi Liu • Shiqing Ma • Yousra Aafer • Wen-Chuang Lee • Juan Zhai		X	X			X	X			
127	Evaluating Input Perturbation Methods for Interpreting CNNs and Saliency Map Comparison	https://arxiv.org/pdf/2101.10977.pdf	2021	• Lukas Brunke • Prateek Agrawal • Nikhil George		X					X			
128	Capsule Network is Not More Robust than Convolutional Network	https://arxiv.org/pdf/2103.15459.pdf	2021	• Jindong Gu • Volker Tresp • Han Hu		X					X			
129	MagFace: A Universal Representation for Face Recognition and Quality Assessment	https://arxiv.org/pdf/2103.06627.pdf	2021	• Qiang Meng • Shichao Zhao • Zhida Huang • Feng Zhou		X						X		
130	Merge and Label: A novel neural network architecture for nested NER	https://www.aclweb.org/anthology/P19-1585.pdf	2019	• Joseph Fisher • Andreas Vlachos			X					X		
131	SampleRNN: An Unconditional End-to-End Neural Audio Generation Model	https://arxiv.org/pdf/1612.07837.pdf	2017	• Soroush Mehrir • Kundan Kumarrr • Ishaan Gulrajanir • Rithesh Kumarr • Shubham Jainr • Jose Sotelor • Aaron Courviller • Yoshua Bengio				X						
132	Generating Black Metal and Math Rock: Beyond Bach, Beethoven, and Beatles	https://arxiv.org/pdf/1811.06639.pdf	2018	• Zack Zukowski • CJ Carr				X						

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
133	Binary Neural Networks: A Survey	https://arxiv.org/pdf/2004.03333.pdf	2020	• Haotong Qina • Ruihao Gonga • Xianglong Liu • Xiao Baie • Jingkuan Songc • Nicu Sebed		X					X			
134	Towards Accurate Binary Convolutional Neural Network	https://arxiv.org/pdf/1711.11294.pdf	2017	• Xiaofan Lin • Cong Zhao • Wei Pan		X					X			
135	Bayesian GAN	https://arxiv.org/pdf/1705.09558.pdf	2017	• Yunus Saatchi • Andrew Gordon Wilson	X	X	X	X	X					
136	A Comparison of ARIMA and LSTM in Forecasting Time Series	https://par.nsf.gov/servlets/purl/10186768	2018	• Neda Tavakoli • Sima Siami-Namini • Akbar Siami Namin				X		X				
137	Proximal Policy Optimisation Algorithms	https://arxiv.org/pdf/1707.06347.pdf	2017	• John Schulman •Filip Wolski •Prafulla Dhariwal, •Alec Radford, •Oleg Klimov										X
138	Ethics Guidelines for Trustworthy AI	https://op.europa.eu/en/publication-detail/-/publication/d3988569-0434-11ea-8c1f-01aa75ed71a1	2019											
139	Evasion Attacks against Machine Learning at Test time	https://link.springer.com/content/pdf/10.1007%2F978-3-642-40994-3_25.pdf	2017	• Battista BiggioIgino • Igino Corona • Davide Maiorca • Blaine Nelson • Nedim Srndie • Pavel Laskov • Giorgio Giacinto • Fabio Roli		X					X			
140	Robustness Evaluations of Sustainable Machine Learning Models against Data Poisoning Attacks in the Internet of Things	https://www.mdpi.com/2071-1050/12/16/6434/htm	2020	• Corey Dunn • Nour Moustafa • Benjamin Turnbull					X		X			
141	Defending network intrusion detection systems against adversarial evasion attacks	https://www.sciencedirect.com/science/article/abs/pii/S0167739X20303368	2020	• Marek Pawlicki • Michał Choraś • Rafał Kozik										
142	Defending SVMs against Poisoning Attacks: the Hardness and DBSCAN Approach	https://arxiv.org/pdf/2006.07757.pdf	2021	• Hu Ding • Fan Yang • Jiawei Huang		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
143	Bagging classifiers for fighting poisoning attacks in adversarial classification tasks	http://pralab.diee.unica.it/sites/default/files/biggio11-mcs.pdf	2011	• Battista Biggio • Igino Corona • Giorgio Fumera • Giorgio Giacinto • Fabio Roli			X		X		X			
144	Adversarial Patch	https://arxiv.org/pdf/1712.09665.pdf	2018	• Tom B. Brown • Dandelion Mané • Aurko Roy • Martín Abadi • Justin Gilmer		X					X			
145	Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures	https://dl.acm.org/doi/pdf/10.1145/2810103.2813677	2015	• Matt Fredrikson • Somesh Jha • Thomas Ristenpart		X	X				X			
146	Membership Inference Attacks against Machine Learning Models	https://arxiv.org/pdf/1610.05820.pdf	2017	• Reza Shokri • Marco Stronati • Congzheng Song • Vitaly Shmatikov		X			X	X	X			
147	STRIDE-AI: An Approach to Identifying Vulnerabilities of Machine Learning Assets	https://github.com/LaraMauri/STRIDE-AI	2021	• Lara Mauri • Ernesto Damiani										
148	For a meaningful Artificial Intelligence	https://www.ai4eu.eu/news/meaningful-artificial-intelligencetowards-french-artificial-and-european-strategy	2018	• Cedric Villani										
149	Strategic Action Plan for Artificial Intelligence	https://www.ai4eu.eu/news/strategic-action-plan-artificial-intelligence-netherlands	2019											
150	On Data Augmentation and Adversarial risk: An empirical Analysis	https://arxiv.org/pdf/2007.02650.pdf	2020	• Hamid Eghdal-zadeh • Khaled Koutini • Paul Primus • Verena Haunschmid • Michal Lewandowski • Werner Zellinger • Bernhard A.Moser • Gerhard Widmer		X					X			
151	Review of Deep Learning Algorithms and Architectures	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8694781	2019	• Ajay Shrestha • Ausif Mahmood										

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
152	Learning in a Large Function Space: Privacy-Preserving Mechanisms for SVM Learning	https://arxiv.org/pdf/0911.5708.pdf	2009	• Benjamin I. P. Rubinstein • Peter L. Bartlett •Ling Huang • Nina Taft							X			
153	Improving Robustness of ML Classifiers against Realizable Evasion Attacks Using Conserved Features	https://arxiv.org/pdf/1708.08327.pdf	2019	• Liang Tong • Bo Li • Chen Hajaj • Chaowei Xiao • Ning Zhang • Yevgeniy Vorobeychik					X		X			
154	Spatially Transformed Adversarial Examples	https://arxiv.org/pdf/1801.02612.pdf	2018	• Chaowei Xiao • Jun-Yan Zhu • Bo Li • Warren He • Mingyan Liu • Dawn Song		X					X			
155	Exploring the Space of Black-box Attacks on Deep Neural Networks	https://arxiv.org/pdf/1712.09491.pdf	2015	• Arjun Nitin Bhagoji • Warren He • Bo Li • Dawn Song		X					X			
156	Robust Physical-World Attacks on Deep Learning Visual Classification	https://arxiv.org/pdf/1707.08945.pdf	2018	• Kevin Eykholt • Ivan Evtimov • Earlence Fernandes • Bo Li • Amir Rahmati • Chaowei Xiao • Atul Prakash • Tadayoshi Kohno • Dawn Song		X					X			
157	Hybrid Isolation Forest - Application to Intrusion Detection	https://arxiv.org/pdf/1705.03800.pdf	2017	• PIERRE-FRANÇCOIS MARTEAU • SAEID SOHEILY-KHAH • NICOLAS BÉCHET	X	X	X	X	X		X			
158	The Performance of LSTM and BiLSTM in Forecasting Time Series	https://par.nsf.gov/servlets/purl/10186554	2019	• Neda Tavakoli • Sima Siami-Namini • Akbar Siami Namin										
159	Anomaly Detection for Data Streams Based on Isolation Forest using Scikit-multiflow	https://hal.archives-ouvertes.fr/hal-02874869/document	2020	• Maurras Togbe • Mariam Barry • Aliou Boly • Yousra Chabchoub • Raja Chiky • Jacob Montiel • Vinh-Thuy Tran					X			X	X	

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
160	Robust Differentiable SVD	https://arxiv.org/pdf/2104.03821.pdf	2021	• Wei Wang • Zheng Dang • Yinlin Hu • Pascal Fua										
161	Accurate Stock Price Forecasting Using Robust and Optimiser Deep Learning Models	https://arxiv.org/ftp/arxiv/papers/2103/2103.15096.pdf	2021	• Jaydip Sen • Sidra Mehtab				X		X				
162	Vulnerabilities of ConnectioNIST 800-53 AI Applications: Evaluation and Defence	https://arxiv.org/pdf/2003.08837.pdf	2020	• Christian Berghoff • Matthias Neu • Arndt Von Twickel		X	X	X	X	X	X			
163	LiBRe: A Practical Bayesian Approach to Adversarial Detection	https://arxiv.org/pdf/2103.14835.pdf	2021	• Zhijie Deng • Xiao Yang • Shizhen Xu • Hang Su • Jun Zhu		X					X			
164	Towards Auditable AI Systems	https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/KI/Towards_Auditable_AI_Systems.pdf?__blob=publicationFile&v=4	2021	• Christian Berghoff, • Battista Biggio • Elisa Brummel • Vasilios Danos • Thomas Doms • Heiko Ehrich • Thorsten Gantevoort • Barbara Hammer • Joachim Iden • Sven Jacob • Heidy Khlaaf • Lars Komrowski • Robert Kröwing • Jan Hendrik Metzen • Matthias Neu • Fabian Petsch • Maximilian Poretschkin • Wojciech Samek • Hendrik Schäbe • Arndt von Twickel • Martin Vechev • Thomas Wiegand										
165	Maxout network	https://arxiv.org/pdf/1302.4389.pdf	2013	• Ian J. Goodfellow • David Warde-Farley • Mehdi Mirza • Aaron Courville • Yoshua Bengio		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
166	A Taxonomy of ML Attacks	https://berryvilleiml.com/taxonomy/	2019	• Victor Shepardson • Gary McGraw • Harold Figueroa • Richie Bonett										
167	BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain	https://arxiv.org/pdf/1708.06733.pdf	2019	• Tianyu Gu • Brendan Dolan-Gavitt • Siddharth Garg		X					X			
168	AI Security and Adversarial Machine Learning 101	https://towardsdatascience.com/ai-and-ml-security-101-6af8026675ff	2019	• Alex Polyakov										
169	Generative Adversarial Networks in Security: A Survey	https://www.researchgate.net/publication/344519514_Generative_Adversarial_Networks_in_Security_A_Survey	2020	• Indira Kalyan Dutta • Bhaskar Ghosh • Michael Totaro • Albert H. Carlson										
170	Secure, Robust and transparent application of AI	https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/KI/Secure_robust_and_transparent_application_of_AI.pdf	2021	• Bundesamt für Sicherheit in der Informationstechnik										
171	Detecting Backdoor Attacks on Deep Neural Networks by Activation Clustering	http://ceur-ws.org/Vol-2301/paper_18.pdf	2018	• Bryant Chen • Wilka Carvalho • Nathalie Baracaldo • Heiko Ludwig • Benjamin Edwards • Taesung Lee • Ian Molloy • Biplav Srivastava	X	X					X			
172	Detection of adversarial training examples in Poisoning Attacks Trough Anomaly Detection	https://arxiv.org/pdf/1802.03041.pdf	2018	• Andrea Paudice • Luis Munoz-Gonzalez • Andras Gyorgy • Emil C.Lupu		X			X		X			
173	When AI Misjudgement is not an accident	https://blogs.scientificamerican.com/observations/when-ai-misjudgment-is-not-an-accident/	2018	• Douglas Yeung										
174	Adversarial attacks on medical machine learning	https://www.media.mit.edu/publications/adversarial-attacks-on-medical-machine-learning/	2029	• Samuel G.Finlayson • Jonathan Zittrain •Joi Ito • Andrew L.Beam • Isaac S.Kohane										

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
175	Camouflaged grafiti on road signs can fool machine learning models	https://thenewstack.io/camouflaged-graffiti-road-signs-can-fool-machine-learning-models/	2017	• Kimberley Mok		X					X			
176	Deepfakes and deep fraud: the new security challenge of misinformation and impersonation	https://www.idgconnect.com/article/3583356/deepfakes-and-deep-fraud-the-new-security-challenge-of-misinformation-and-impersonation.html	2020	• Sadia Sajiad										
177	Stealing Hyperparameters in Machine Learning	https://arxiv.org/pdf/1802.05351.pdf	2019	• B.Wang • N.Z.Gong		X			X	X	X			
178	Explanation-Guided Backdoor Poisoning Attacks Against Malware Classifiers	https://www.usenix.org/system/files/sec21-severi.pdf	2020	• Giorgio Severi • Jim Meyer • Scott Coull • Alina Oprea					X		X			
179	Adversarial Examples that Fool both Computer Vision and Time-Limited Humans	//arxiv.org/pdf/1802.08195.pd	2018	• Gamaleldin F. Elsayed • Shreya Shankar • Brian Cheung • Nicolas Papernot • Alex Kurakin • Ian Goodfellow • Jascha Sohl-Dickstein		X					X			
180	With Great training comes great vulnerability: Practical Attacks against transfer learning	https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-wang.pdf	2018	• Bolun Wang • Bimal Viswanath • Yuanshun Yao • Haitao Zheng • Ben Y.Zhao										
181	Practical black-box attacks against machine learning	https://arxiv.org/pdf/1602.02697.pdf	2017	• Nicolas Papernot • Patrick McDaniel • Ian Goodfellow • Somesh Jha • Z. Berkay Celik • Ananthram Swami		X					X			
182	Universal adversarial perturbations	https://arxiv.org/pdf/1610.08401.pdf	2017	• Seyed-Mohsen Moosavi-Dezfooli • Alhussein Fawzi • Omar Fawzi • Pascal Frossard		X					X			
183	Benchmarking neural network robustness to common corruptions and perturbations	https://arxiv.org/pdf/1903.12261.pdf	2019	• Dan Hendrycks • Thomas Dietterich		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
184	Securing Artificial Intelligence, Part 1 The attack surface of machine learning and its implications	https://www.researchgate.net/publication/341792988_Securing_Artificial_Intelligence_Part_1_The_attack_surface_of_machine_learning_and_its_implications/link/5ed5064a299bf1c67d3238f4/download	2019	• Sven Herping										
185	Support vector machines under adversarial label noise	http://proceedings.mlr.press/v20/biggio11/biggio11.pdf	2011	• Battista Biggio • Blaine Nelson • Pavel Laskov		X			X		X			
186	Just How Toxic is Data Poisoning? A Unified Benchmark for Backdoor and Data Poisoning Attacks	https://arxiv.org/pdf/2006.12557.pdf	2020	• Avi Schwarzschild • Micah Golddblum • Arjun Gupta • John P.Dickerson • Tom Goldstein		X					X			
187	Is feature selection secure against training data poisoning?	https://arxiv.org/pdf/1804.07933.pdf	2018	• Huang Xiao • Battista Biggio • Gavin Brown • Giorgio Fumera • Claudia Eckert • Fabio Roli										
188	You Autocomplete Me: Poisoning Vulnerabilities in Neural Code Completion	http://pages.cs.wisc.edu/~jerryzhu/ssl/pub/Mei2015Machine.pdf	2021	• Roei Schuster • Congzheng Song • Eran Tromer • Vitaly Shmatikov			X					X		
189	Model-reuse attacks on deep learning systems	https://arxiv.org/pdf/1812.00483.pdf	2018	• Yujie Ji • Xinyang Zhang • Shouling Ji • Xiapu Luo • Ting Wang		X		X	X	X	X			
190	A survey on transfer learning	https://ieeexplore.ieee.org/document/5288526	2009	• Sinno Jialin Pan • Qiang Yang										
191	Sponge examples: Energy-Latency Attacks on Neural Network	https://arxiv.org/pdf/2006.03463.pdf	2021	• Ilia Shumailov • Yiren Zhao • Daniel Bates • Nicolas Papernot • Robert Mullins • Ross Anderson		X					X			
192	. Impact analysis of false data injection attacks on power system static security assessment	https://link.springer.com/content/pdf/10.1007%2Fs40565-016-0223-6.pdf	2016	• Jiongcong Chen • Gaoqi Liang • Zexiang CAI • Chunchao Hu • Yan Xu • Fengji Luo • Junhua Zhao										

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
193	The ND2DB attack: Database content extraction using timing attacks on the indexing algorithms	https://www.researchgate.net/publication/250195790_The_ND2DB_attack_Database_content_extraction_using_timing_attacks_on_the_indexing_algorithms	2007	• Ariel Futoransky • Damian Saura • Ariel Waissbein				X						
194	La nouvelle technologie de protection des données		2020	• Théo Ryffel										
195	Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks	https://arxiv.org/pdf/2003.01690.pdf	2020	• Francesco Croce • Matthias Hein		X					X			
196	Manipulating Machine Learning: Poisoning attacks and countermeasures for regression learning	https://arxiv.org/pdf/1804.00308.pdf	2021	• M.Jagielski • Alina Oprea • Battista Biggio • Chang Liu • Cristina Nita-Rotaru • Bo Li					X	X				
197	Using Machine teaching to identify Optimal training-set attacks on Machine Learners	http://pages.cs.wisc.edu/~jerryzhu/machineteaching/pub/Mei2015Machine.pdf	2015	• Shike Mei • Xiaojin Zhu					X	X	X			
198	STRIP: A defence against trojan attacks on deep neural networks	https://arxiv.org/pdf/1902.06531.pdf	2020	• Yansong Gao • Chang Xu • Derui Wang • Shiping Chen • Damith C. Ranasinghe •Surya Nepal		X					X			
199	Misleading learners: Co-opting your spam filter	https://people.eecs.berkeley.edu/~tygar/papers/SML/misleading.learners.pdf	2009	• Blaine Nelson • Marco Barreno • Fuching Jack Chi • Anthony D.Joseph • Benjamin I.P Rubinstein •Udam Saini • Charles Sutton • J.D Tygar •Kai Xia					X		X			
200	Cascade Adversarial Machine learning Regulariser with a unified embedding	https://arxiv.org/pdf/1708.02582.pdf	2018	• Taesik Na • Jong Hwan Ko • Saibal Mukhopadhyay		X					X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
201	Gradient band-based adversarial training for generaliser attack immunity of A3C path finding.	https://arxiv.org/pdf/1807.06752.pdf	2018	• Tong Chen • Wenjia Niu • Yingxiao Xiang • Xiaoxuan Bai • Jiqiang Liu •Zhen Han •Gang Li		X								X
202	Certifying Some Distributional Robustness with Principled Adversarial Training	https://arxiv.org/pdf/1710.10571.pdf	2020	• Adam Sinha • Hongsoek Namkoong • Riccardo Volpi • John Duchi		X					X			
203	Differentially Private Empircal Risk Minimisation	https://www.jmlr.org/papers/volume12/chaudhuri11a/chaudhuri11a.pdf	2011	• Kamalika Chaudhuri • Claire Monteleoni • Anand D.Sarwate					X		X			
204	A Taxonomy and Terminology of Adversarial Machine Learning	https://csrc.nist.gov/publications/detail/nistir/8269/draft	2019	•Elham Tabassi •Kevin J. Burns •Michael Hadjimichael •Andres D. Molina-Markham •Julian T. Sexton										
205	Deep Defense: Training DNNs with improved adversarial Robustness	https://arxiv.org/pdf/1803.00404.pdf	2018	• Ziang Yan • Yiwen Guo • Changshui Zhang		X					X			
206	Improving the Adversarial Robustness and Interpretability of Deep Neural Network by Regularizing their input Gradients	https://arxiv.org/pdf/1711.09404.pdf	2017	• Andrew Slavin Ross • Finale Doshi-Velez		X					X			
207	On detecting Adversarial perturbation	https://arxiv.org/pdf/1702.04267.pdf	2017	• Jan Hendrik Metzen • Tim Genewein • Volker Fischer • Bastian Bischoff		X					X			
208	SoK: Security and privacy in machine learning	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8406613	2017	• Nicolas Papernot • Patrick McDaniel • Arunesh Sinha • Michael P. Wellman										
209	Random Feature Nullification for Adversary Resistant Deep Architecture	https://arxiv.org/pdf/1610.01239v1.pdf	2016	• Qinglong Wang • Wenbo Guo • Kaixuan Zhang • Xinyu Xing • C. Lee Giles • Xue Liu		X					X			
210	The security of machine learning	https://people.eecs.berkeley.edu/~adj/publications/paper-files/SecML-MLJ2010.pdf	2010	• Marco Barreno • Blaine Nelson • Anthony D. Joseph			X				X			

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
211	Ensemble adversarial training: attacks and defenses	https://arxiv.org/pdf/1705.07204.pdf	2018	• Florian Tramer • Alexey Kurakin • Nicolas Papernot • Ian Goodfellow • Dan Boneh • Patrick McDaniel		X					X			
212	Improving the robustness of Deep Neural Networks via Stability training	https://arxiv.org/pdf/1604.04326.pdf	2016	• Stephen Zheng • Yang Song • Thomas Leung • Ian Goodfellow		X					X			
213	Distillation as a defense to adversarial perturbations against deep neural networks	https://arxiv.org/pdf/1511.04508.pdf	2016	• Nicolas Papernot • Patrick McDaniel • Xi Wu • Somesh Jha • Ananthram Swami		X					X			
214	Scalable Private Learning with Pate	https://arxiv.org/pdf/1802.08908.pdf	2016	• Nicolas Papernot • Shuang Song • Ilya Mironov • Ananth Raghunathan • Kunal Talwar • Ulfar Erlingsson		X					X			
215	Gradient masking in machine learning	https://seclab.stanford.edu/AdvML2017/slides/17-09-aro-aml.pdf	2017	• Nicolas Papernot										
216	The Quest for Statistical Significance: Ignorance, Bias and Malpractice of Research Practitioners	https://hal.archives-ouvertes.fr/hal-01758493/document	2018	•Joshua Abah										
217	The measure and mismeasure of fairness A critical review of fair machine learning	https://arxiv.org/pdf/1808.00023.pdf	2018	• Sam Corbett-Davies • Sharad Goel										
218	Adversarial Policy Training Against Deep Reinforcement Learning	https://www.usenix.org/system/files/sec21-wu-xian.pdf	2021	• Xian Wu • Wenbo Guo • Hua Wei • Xinyu Xing		X			X					X
219	Early stopping - But When?	https://link.springer.com/book/10.1007%2F978-3-642-35289-8	2012	• Lutz Prechelt										
220	A pitfal and solution in multi-class feature selection for text classification	https://icml.cc/Conferences/2004/proceedings/papers/107.pdf	2004	• George Forman			X					X		
221	Near-Optimal Algorithms for Differentially-Private principal components	https://arxiv.org/pdf/1207.2812.pdf	2013	• Kamalika Chaudhuri • Anand D. Sarwate • Kaushik Sinha					X				X	

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
222	Differentially private model selection via Stability argument and the robustness of the Lasso	http://proceedings.mlr.press/v30/Guha13.pdf	2013	• Adam Smith • Abhradeep Thakurta						X				
223	WaveGuard: Understanding and Mitigating Audio Adversarial Examples	https://www.usenix.org/system/files/sec21-hussain.pdf	2021	• Shehzeen Hussain • Paarth Neekhara • Shlomo Dubnov • Julian McAuley, • Farinaz Koushanfar				X			X			
224	Privacy Preserving Machine Learning	http://researchers.lille.inria.fr/abellet/teaching/private_machine_learning_course.html	2020	•Aurélien Bellet										
225	Interpretability of Machine Learning What are the challenges in the era of automated decision-Making Progresses?	https://www.wavestone.com/app/uploads/2019/09/Wavestone_Interpretability_machine_learning.pdf	2019	•Alexandre Vérine •Stephan Mir										
226	6 Python Libraries to interpret Machine Learning Models and Build Trust	https://www.analyticsvidhya.com/blog/2020/03/6-python-libraries-interpret-machine-learning-models/	2020	•Purva Huilgol										
227	Pitfalls to avoid when Interpreting Machine Learning Models	https://arxiv.org/pdf/2007.04131.pdf	2020	• Cristoph Molnar • Gunnar König • Julia Herbinger • Timo Freiesleben • Susanne Dandl • Christian A.Scholbeck • Giuseppe Casalicchio • Moritz Grosse-Wentrup • Bernd Bischl										
228	7 steps to ensure and sustain Data quality	https://towardsdatascience.com/7-steps-to-ensure-and-sustain-data-quality-3c0040591366	2019	• Stéphanie Shen										

Index	Title	Source	Publication date	Author	Type of data ingested					Supervised Learning		Unsupervised Learning		Reinforcement learning
					Video	Image	Text	Time series	Structured Data	Regression	Classification	Clustering	Dimension Reduction	Rewarding
229	Three ways to avoid bias in machine learning	https://techcrunch.com/2018/11/06/3-ways-to-avoid-bias-in-machine-learning/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAAIfXhVlDfTYv80Vxw4JVaKFZyt_3_2DTapBQQjW8C1vzjPTQqViKdAE5O-BV1Q5J5waGMcYo4yu2R4QBOr9H17RpApdX9vlXDUlo_MS28Q4GD8qCXqhogX534lJcR7DPrzwANTY8WPnJX5GXVmytlHxM0ZK9Ym2ANzKGSnae6QgH	2018	• Vince Lynch										
230	5 Ways to deal with the lack of data in Machine Learning	https://www.kdnuggets.com/2019/06/5-ways-lack-data-machine-learning.html	2019	•Alexandre Gonfalonieri										
231	Artificial Intelligence is Crucial TO the Success of Your Business and here is why	https://towardsdatascience.com/artificial-intelligence-is-crucial-to-the-success-of-your-business-learn-why-d5b96fa3564d	2019	• Amit Makhija										
232	7 Simple Techniques to Prevent Overfitting	https://www.kaggle.com/learn-forum/157623	2020	•Devendra Kumar Yadav										

TP-06-21-153-EN-N

ABOUT ENISA

The European Union Agency for Cybersecurity, ENISA, is the Union's agency dedicated to achieving a high common level of cybersecurity across Europe. Established in 2004 and strengthened by the EU Cybersecurity Act, the European Union Agency for Cybersecurity contributes to EU cyber policy, enhances the trustworthiness of ICT products, services and processes with cybersecurity certification schemes, cooperates with Member States and EU bodies, and helps Europe prepare for the cyber challenges of tomorrow. Through knowledge sharing, capacity building and awareness raising, the Agency works together with its key stakeholders to strengthen trust in the connected economy, to boost resilience of the Union's infrastructure, and, ultimately, to keep Europe's society and citizens digitally secure. More information about ENISA and its work can be found here: www.enisa.europa.eu.

ENISA
European Union Agency for Cybersecurity

Athens Office
Agamemnonos 14, Chalandri 15231, Attiki, Greece

Heraklion Office
95 Nikolaou Plastira
700 13 Vassilika Vouton, Heraklion, Greece

enisa.europa.eu

ISBN: 978-92-9204-543-2
DOI: 10.2824/874249