Overview of Adversarial Machine Learning and AI Safety

Fabio Massimo Zennaro
fabiomz@ifi.uio.no

University of Oslo

February 7, 2019

Aim and Organization

In this presentation we are going to introduce and reviews aspects of **security** of **machine learning**.

1. Concepts from machine learning
2. Attacks against machine learning [Papernot et al., 2016a; Biggio and Roli, 2018]
3. Defenses for machine learning [Biggio and Roli, 2018; Akhtar and Mian, 2018]
4. Safety of machine learning [Amodei et al., 2016c]

1. Machine Learning

What is machine learning?

ML is the field studying *automated induction procedures to develop useful models*.

- *Automated procedures*: algorithms
- *Induction*: from particular (data) to general (model)
- *Models*: abstractions of a phenomenon [Floridi, 2011]
- *Useful*: allowing us to explain/predict/control [Floridi, 2011]

What is model?

A **model** is a mathematical representation of a phenomenon.

$$f : X \to Y \qquad \begin{matrix} P(X) \\ P(X, Y) \\ P(Y|X) \end{matrix}$$

There are three popular flavours of models (related to three types of learning algorithms):

- *Supervised:* $f : X \to Y$ $\qquad P(Y|X)$
- *Unsupervised:* $f : X \to Z$ $\qquad P(X, Z)$
- *Reinforcement:* $f : \mathcal{S} \to \mathcal{A}$ $\qquad \pi(a|s)$

Lifecycle of a model (I)

There are two main stages in the lifecycle of machine learning

- *Learning*: learning a specific model

$$f : X \rightarrow Y$$

- *Inference* or *deployment*: using the model

$$f : X \rightarrow Y$$

We want the model to **generalize**: learn from *specific* x, infer for *all* x.

Lifecycle of a model (II)

To assess generalization we partition our data into: **training data** (used to learn) and **test data** (used for evaluation)

- *Learning*: learning a specific model from collected data

$$f : X^{tr} \rightarrow Y^{tr}$$

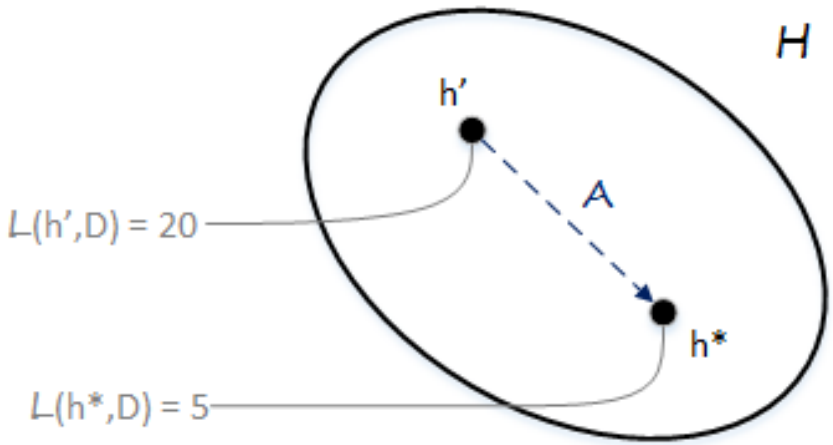
- *Inference* or *deployment*: using the model on never-seen-before data

$$f : X^{te} \rightarrow Y^{te}$$

This is meaningful is training and test data are *independent samples from the same distribution*: $p(X^{tr}) = p(X^{te})$

Learning (I)

1. *Data* $\mathcal{D}$
2. *Family of models* or *hypothesis space* $\mathcal{H}$
3. *Loss/objective/reward function* $\mathcal{L}(h, \mathcal{D})$
4. *Exploration strategy* of the hypothesis space $\mathcal{A}$

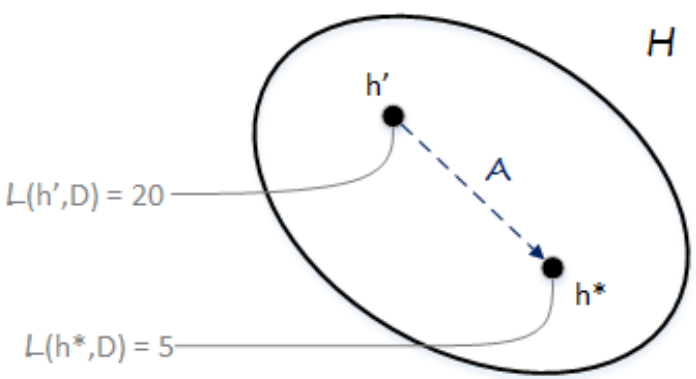


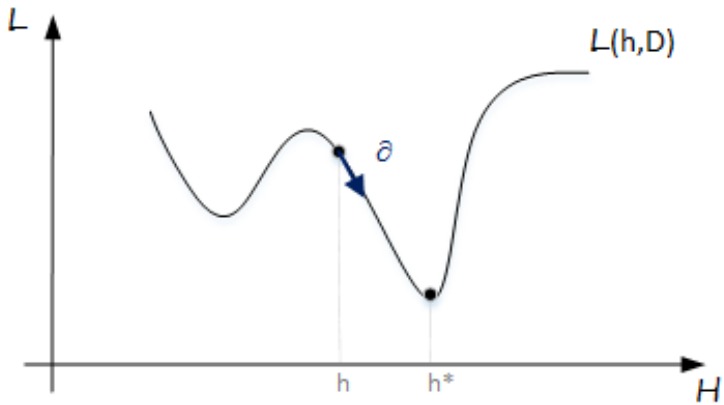
Hypothesis space, loss function and exploration strategy are usually tightly bound and comes as a *machine learning algorithm*.

Learning (II)

Learning means solving an **optimization problem**:

$$h* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \mathcal{L}(h, \mathcal{D})$$





Learning (III)

Example: Learning to discriminate digits using a neural network

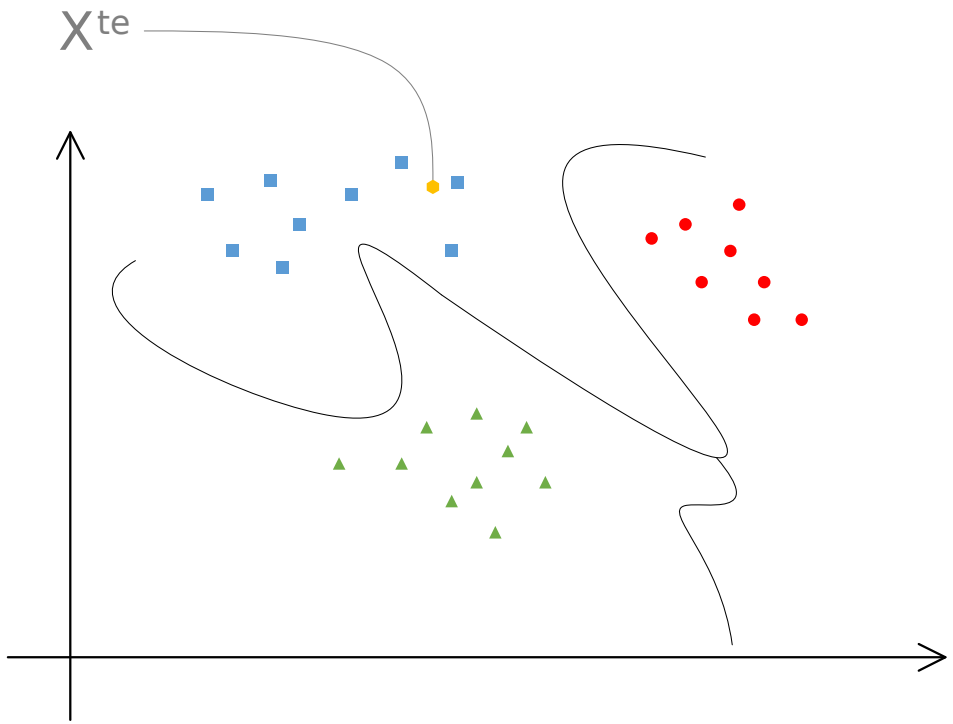
$$f : \text{Image} \rightarrow \text{Label}$$

1. *Data:* $\mathcal{D} = \{\text{Set of digits and labels}\}$
2. *Hypothesis space:* $\mathcal{H} =$ approximate continuous functions on compact subsets of $\mathcal{R}^n$ [Cybenko, 1989]
3. *Loss function:* $\mathcal{L} =$ mean squared error in prediction
4. *Exploration strategy:* $\mathcal{A} =$ gradient descent

$$h* = \underset{h \in \mathcal{H}}{\text{argmin}}\, \mathcal{L}(h, \mathcal{D})$$

Inference (I)

Inference means evaluating the learned function.



Some generic remarks

- There is no thing such *THE* model of the data [Wolpert and Macready, 1997].
- A model must be built on *assumptions* [MacKay, 2003].
- A model is not correct or wrong; it must be properly *evaluated*.
- Only what can be induced from the data can be learned; beware, though, the space not constrained by data.
- There are always *trade-offs* to consider:
  Training performance vs Test performance [Domingos, 2012]
  Expressivity vs Efficiency
  Performance vs Interpretability
  Efficiency vs Security [Tsipras et al., 2018]

2. Attacks against Machine Learning

Adversarial machine learning

Doing machine learning in an *adversarial setting*.

Two main traditions of research on security in machine learning [Biggio and Roli, 2018]:

- *Security of ML (∼2004-2005)*: studying security of ML models in the computer security field [Dalvi et al., 2004];
- *Adversarial ML (∼2014)*: studying security of deep ML models [Szegedy et al., 2013]

Characterizing the threat [Papernot et al., 2016a; Biggio and Roli, 2018]

Attack Time:

- *Learning:* attacking during the learning phase
- *Inference:* attacking during the inference phase

Attacker Goal:

- *Integrity-Availability:* compromise learning or inference
- *Confidentiality-Privacy:* extracting data or information about the model

Attacker Knowledge:

- *White-box knowledge:* perfect knowledge of data and model
- *Gray-box knowledge:* partial knowledge of data and/or model
- *Black-box knowledge:* minimal knowledge of data and/or model

Characterizing the threat [Papernot et al., 2016a; Biggio and Roli, 2018]

Attacker Specificity:

- *Targeted:* aimed at specific effect
- *Indiscriminate:* generally aimed at subversion

Attacker Constraint:

- *Min-perturbation:* given the desired effect, choose the attack that minimize the detectability.
- *Max-confidence:* given the possible perturbation, choose the attack that maximize the effect.

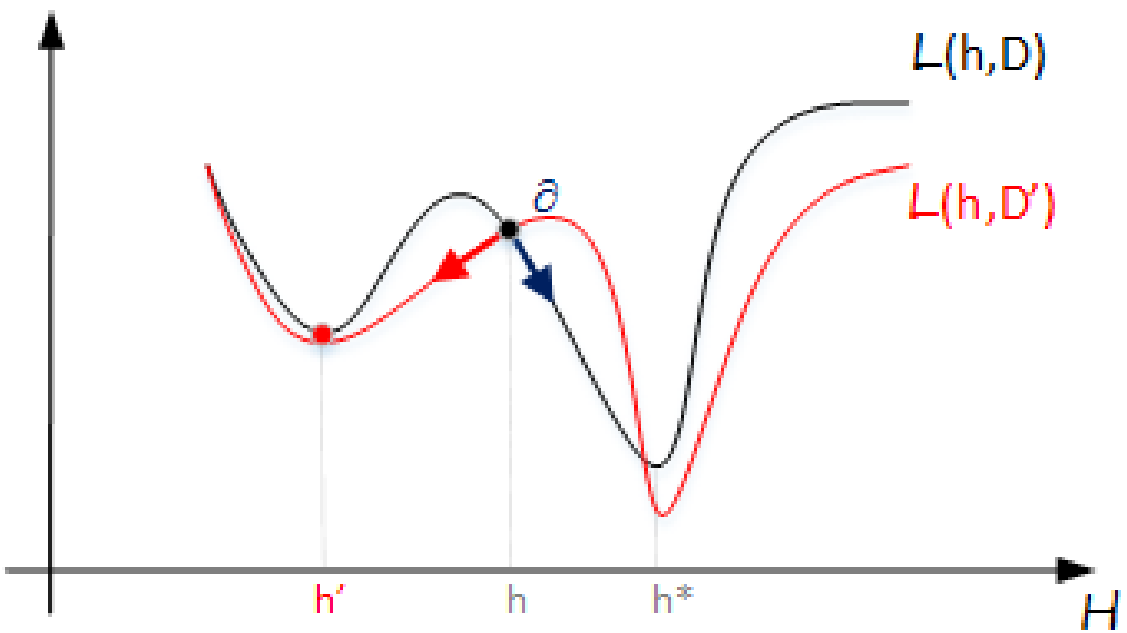
Attack Surface:

- *Data:* collection and processing of data $\mathcal{D}$
- *Model:* including hypothesis space $\mathcal{H}$, loss function $\mathcal{L}$ and learning strategy $\mathcal{A}$

Integrity attacks at learning time

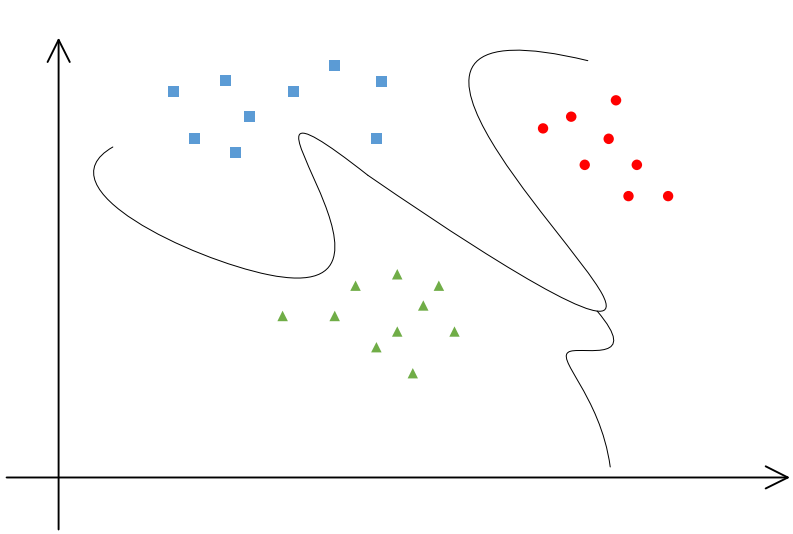
Attacks aimed at derailing learning.

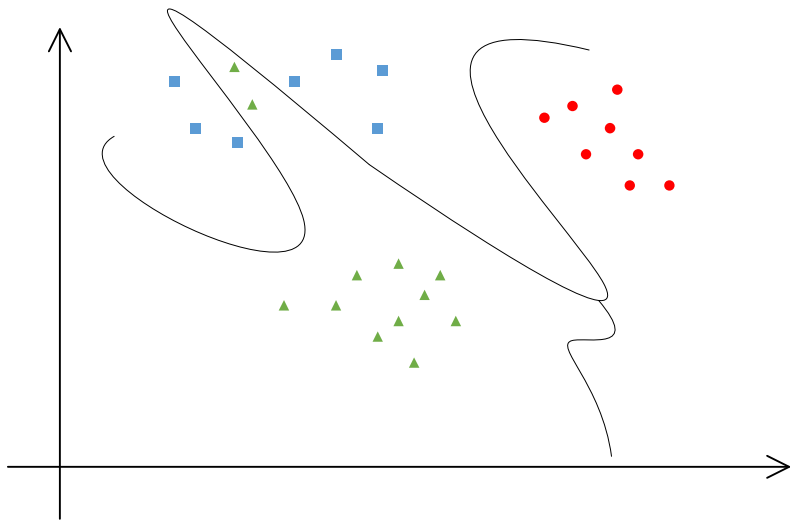
$$\mathcal{D}' = \underset{\mathcal{D}'}{\operatorname{argmin}} \mathcal{L}'\left(h, \mathcal{D}'\right)$$
$$\text{s.t.} \quad h' = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \mathcal{L}\left(h, \mathcal{D} \cup \mathcal{D}'\right)$$



Integrity attacks at learning time

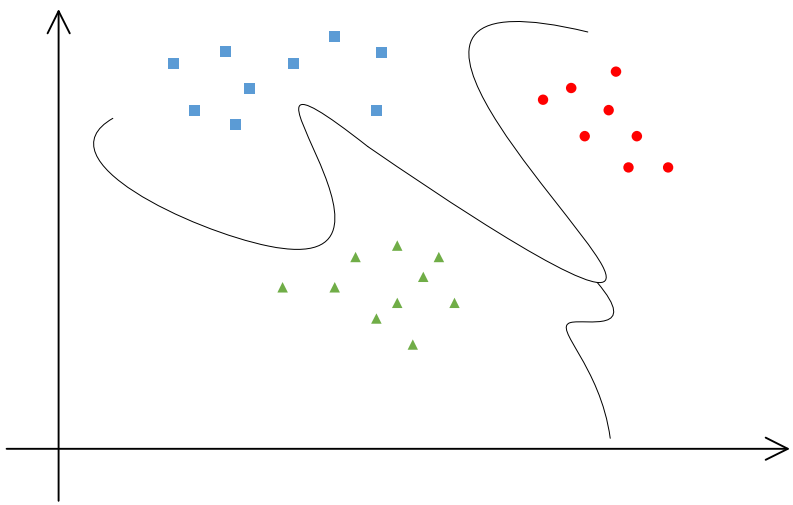
- *Label manipulation:* harmful perturbation of labels given partial or full knowledge of a model [Biggio et al., 2011; Mozaffari-Kermani et al., 2015]

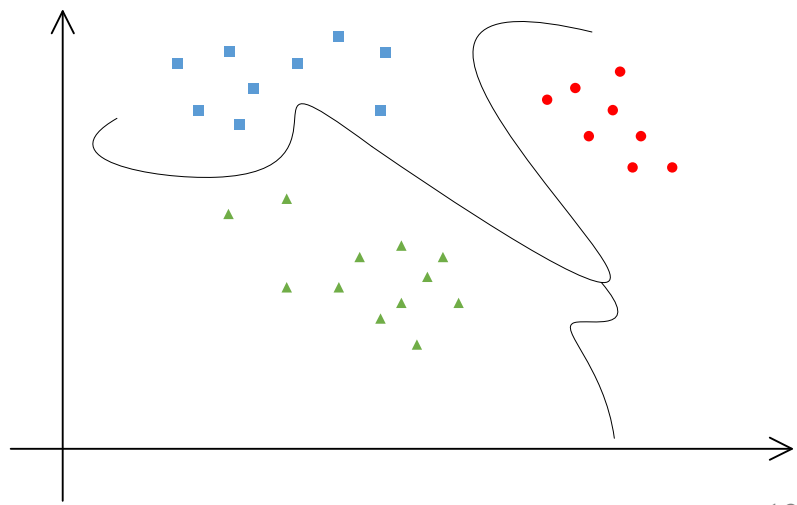




Integrity attacks at learning time

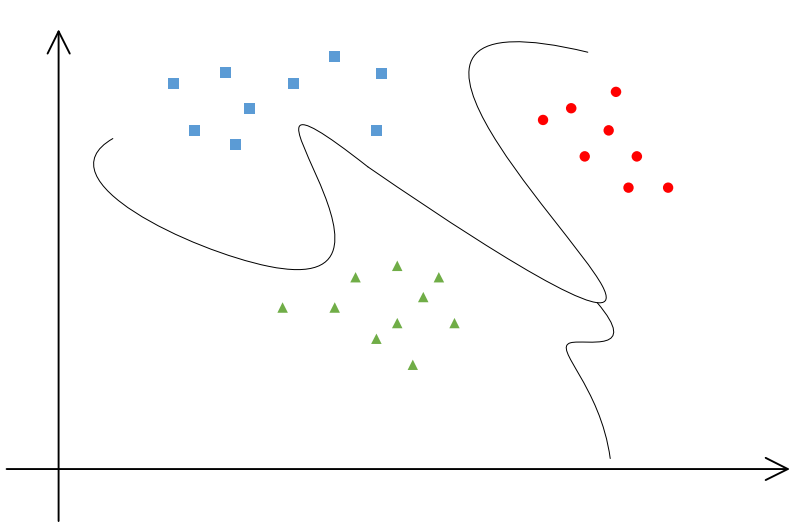
- *Direct data poisoning:* insertion of spurious data points in the data set to compromise learning [Kloft and Laskov, 2010; Mei and Zhu, 2015; Steinhardt et al., 2017]
- *Indirect data poisoning:* malicious modification of the data generating process to generate inconsistent data [Perdisci et al., 2006]

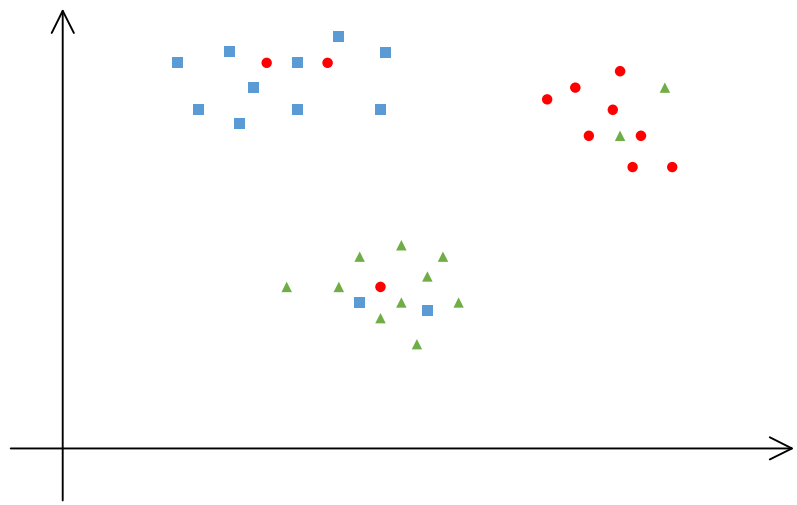




Integrity attacks at learning time

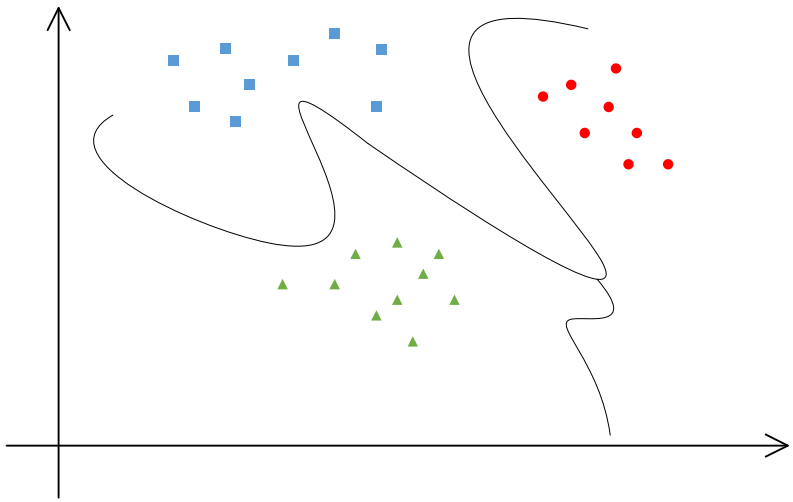
- *Denial:* insertion of random data points to prevent learning.

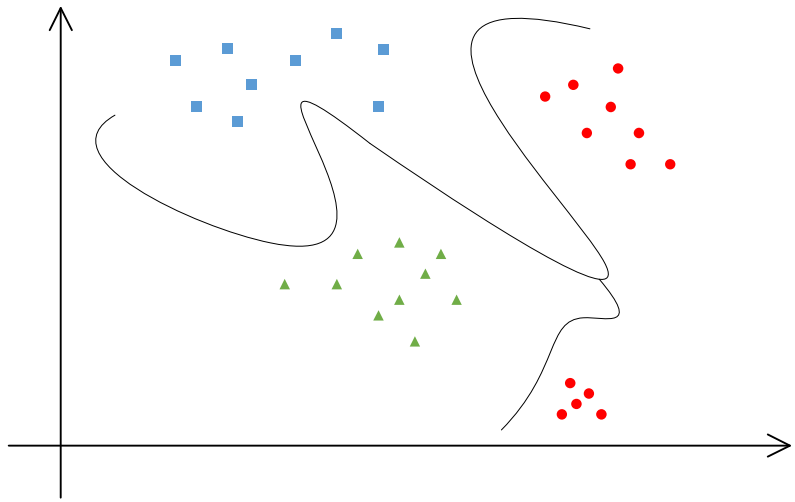




Integrity attacks at learning time

- *Backdoor:* insertion of a signal to misdirect learning [Chen et al., 2017; Gu et al., 2017].





Remarks on attacks at learning time

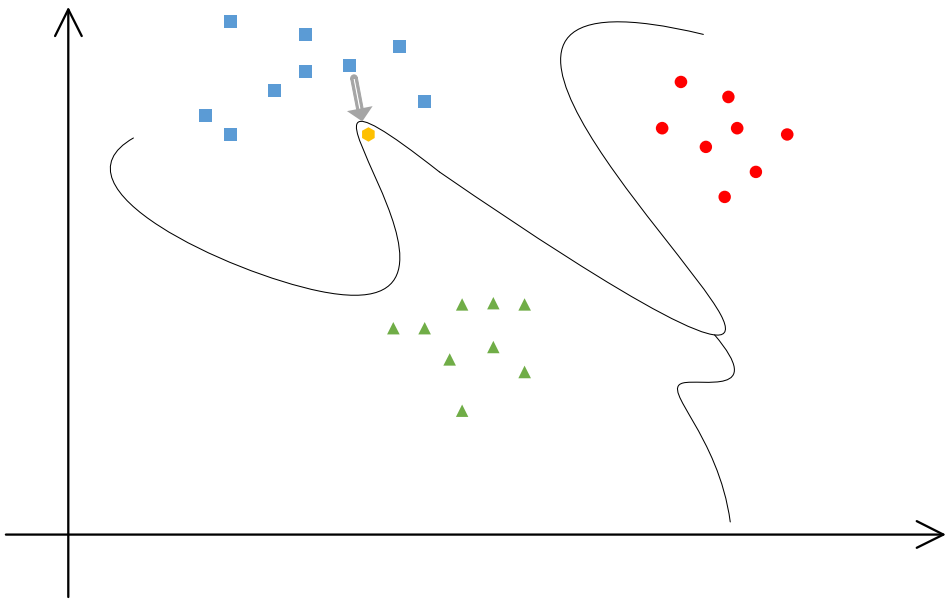
- Data and models are often public.
- Attack at learning time may range from *indiscriminate* (random denial) to *targeted* (extend the domain of a class).
- Optimal poisoning samples can be computed relying on *gradient-based attacks*.
- Learning-time attacks may happen at inference time, if the model keeps learning (Microsoft Tay).
- Attacks cross digitial and real world.

Integrity attacks at inference time

Attacks aimed at fooling the network.

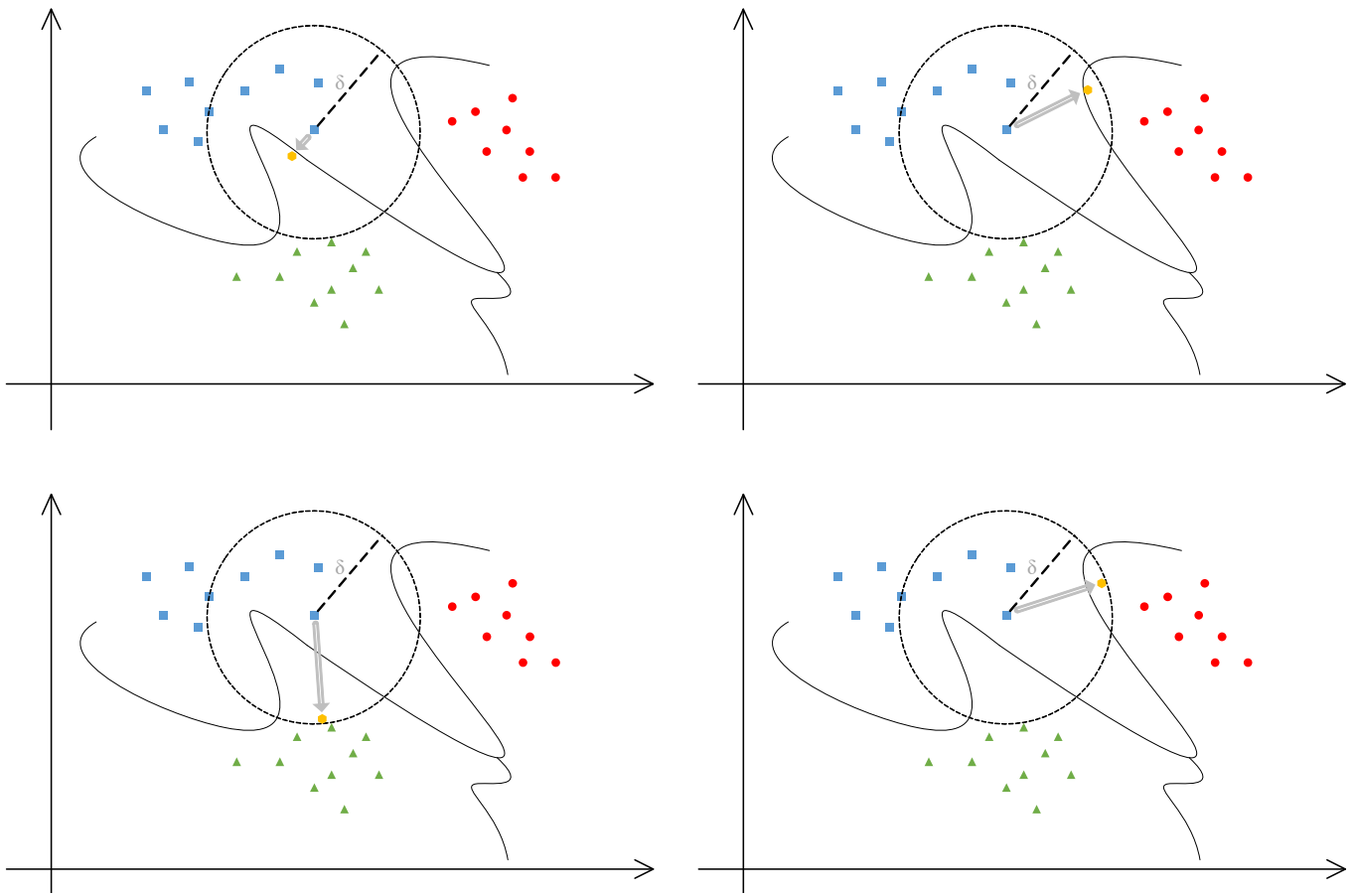
$$\min_{\delta} |\delta|_p \quad \text{s.t.} \quad h*(x) \neq h*(x+\delta)$$

$$\underset{\delta}{\text{argmax}} \, |h*(x) - h*(x+\delta)| \quad \text{s.t.} \quad \delta < M$$



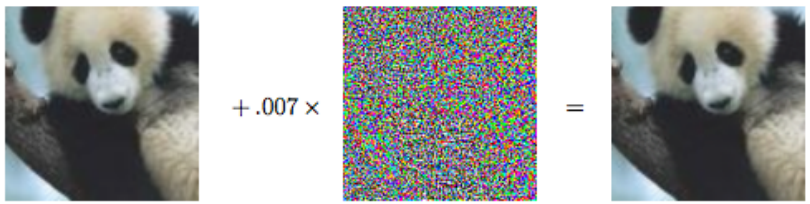
Integrity attacks at inference time

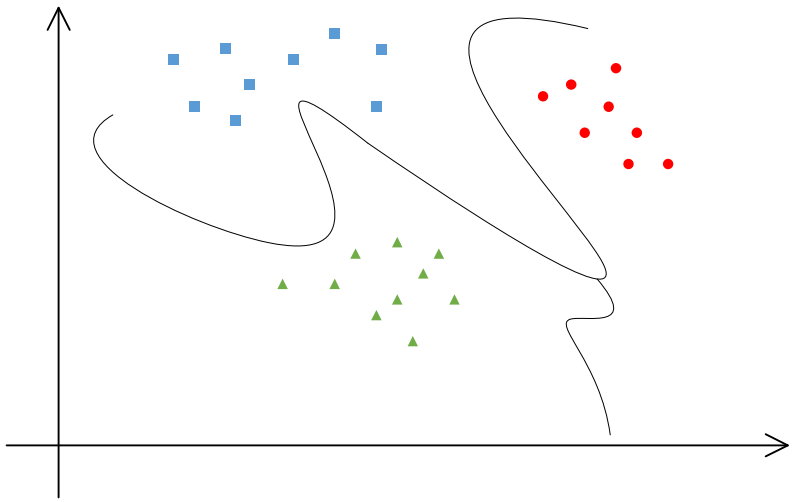
- *Min-perturbation/Max-confidence indiscriminate/targeted (red) adversarial attacks*

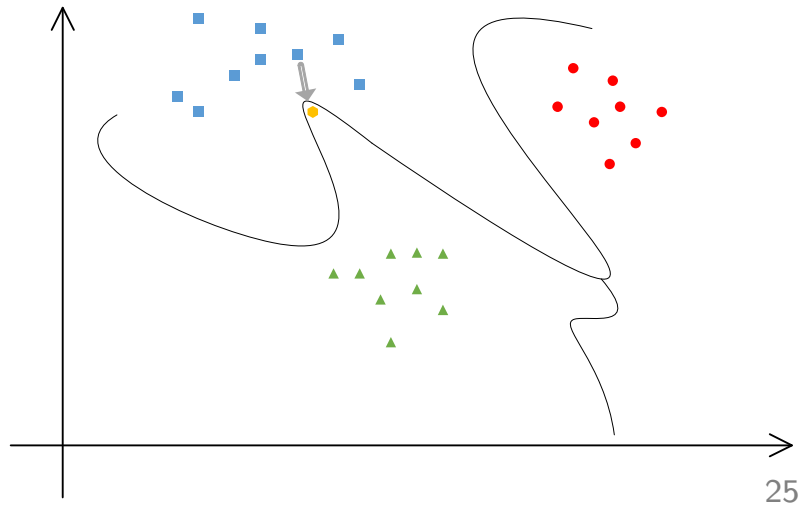


Integrity attacks at inference time

- *Direct poisoning using adversarial examples:* generation of adversarial data points exploiting gradient [Szegedy et al., 2013; Goodfellow et al., 2014]

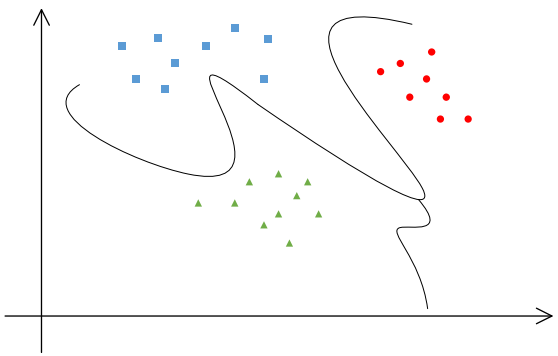


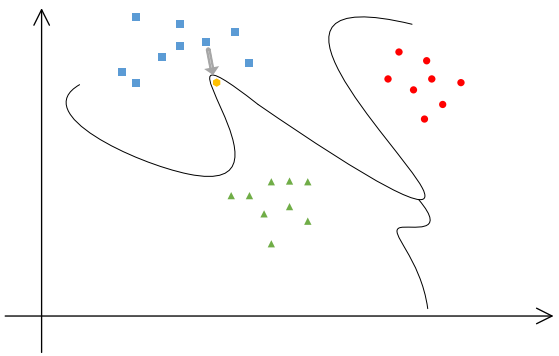




Integrity attacks at inference time

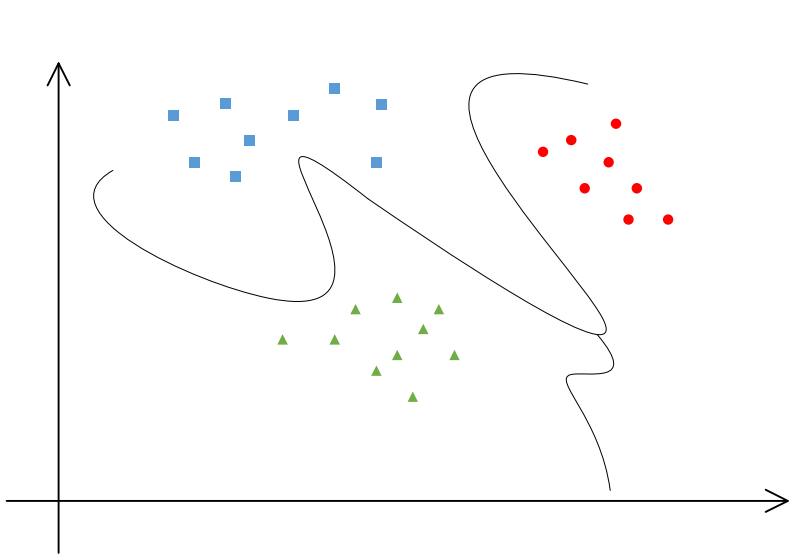
- *Indirect poisoning using adversarial examples:* insertion of adversarial examples in the data processing pipeline [Kurakin et al., 2016]

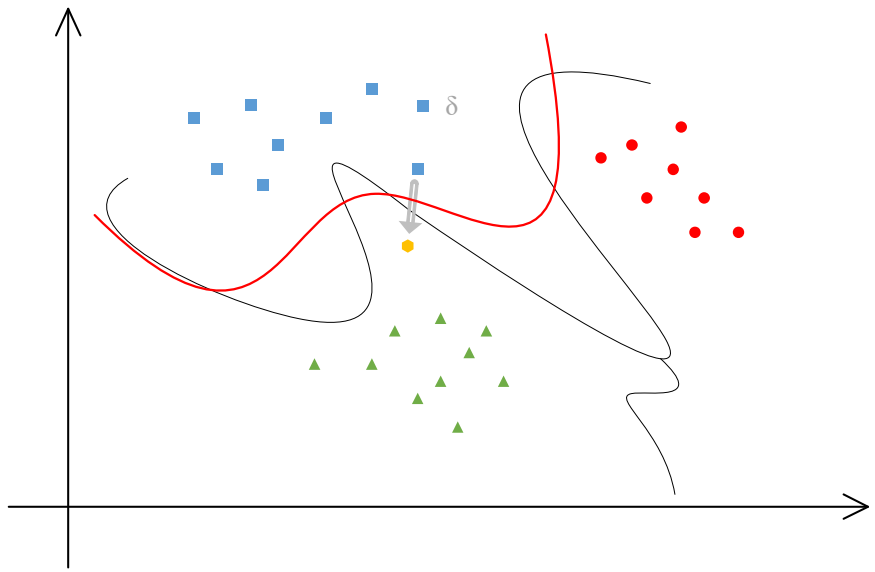




Integrity attacks at inference time

- *Adversarial example transferability:* use of adversarial data points generated on an approximate substitute model [Szegedy et al., 2013]





Privacy attacks at inference time

Attacks aimed at extracting sensitive information.

- *Membership test:* querying the model to discover if specific data points were part of the training set
- *Statistical property test:* querying the model to determine statistical properties of the training set [Ateniese et al., 2015]
- *Model inversion attack:* recovering information about the inputs from the outputs [Fredrikson et al., 2014]
- *Model extraction:* retrieving value of model parameters from outputs [Tramèr et al., 2016]

Remarks on attacks at learning time

- *Gradient-based methods* are used to find adversarial examples for differentiable models (e.g: projected gradient, fast gradient sign, DeepFool, one-pixel genetic modification, universal adversarial perturbations)
- However non-differentiable models are vulnerable too.
- Attacks cross digitial and real world.

3. Defenses for Machine Learning

Characterizing the Defense [Biggio and Roli, 2018; Akhtar and Mian, 2018]

Defense Stance:

- *Reactive:* readily address new attacks
- *Proactive:* plan to prevent future attacks

Defense Paradigm:

- *Detection:* catch new attacks in advance
- *Prevention:* be resistant to attacks

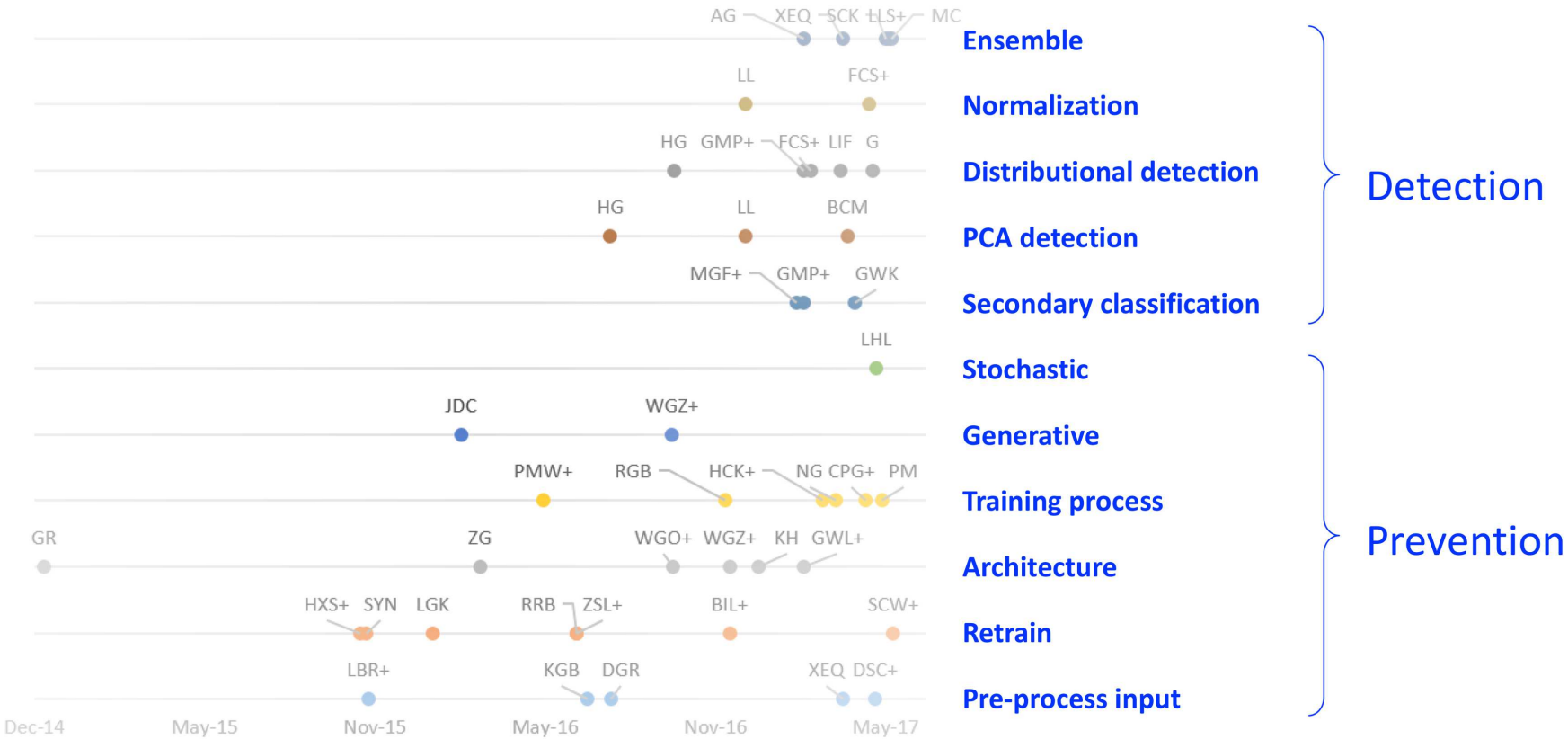
Defense Time:

- *Learning:* protect the learning process
- *Inference:* protect the inference process

Defense Target:

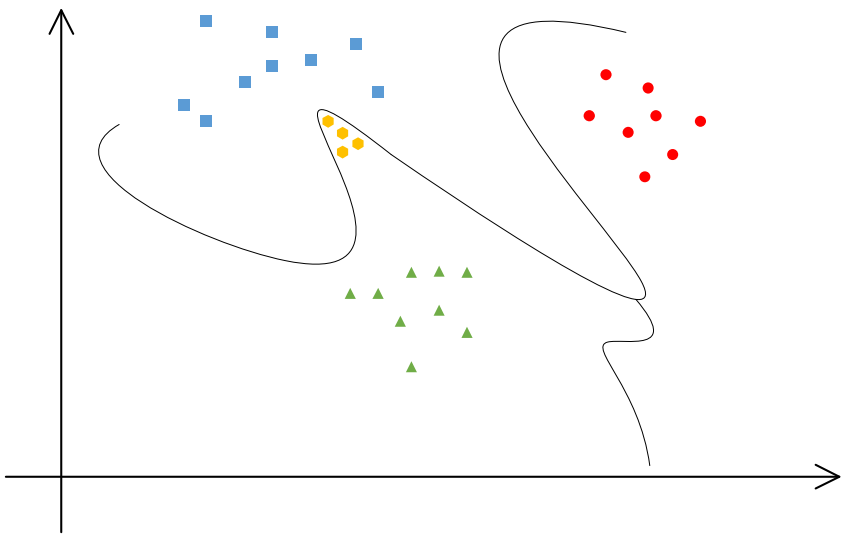
- *Data:* modify the data to increase defense
- *Model:* modify the model to improve robustness
- *Other:* extend the system

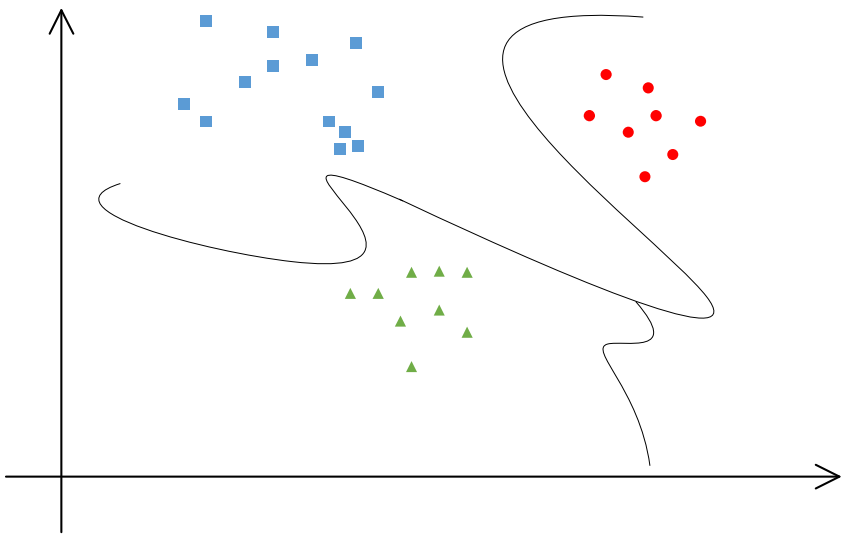
Characterizing the Defense [Song, 2018]



Defense at learning time

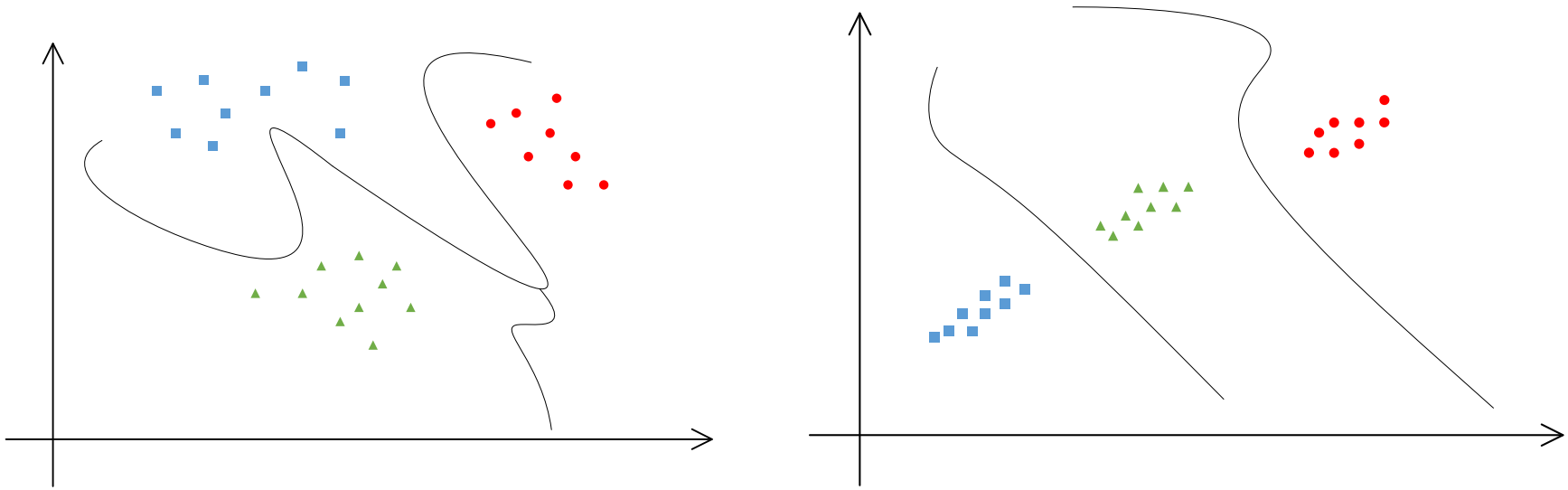
- *Adversarial training:* exploit adversarial samples to improve your model [Szegedy et al., 2013]





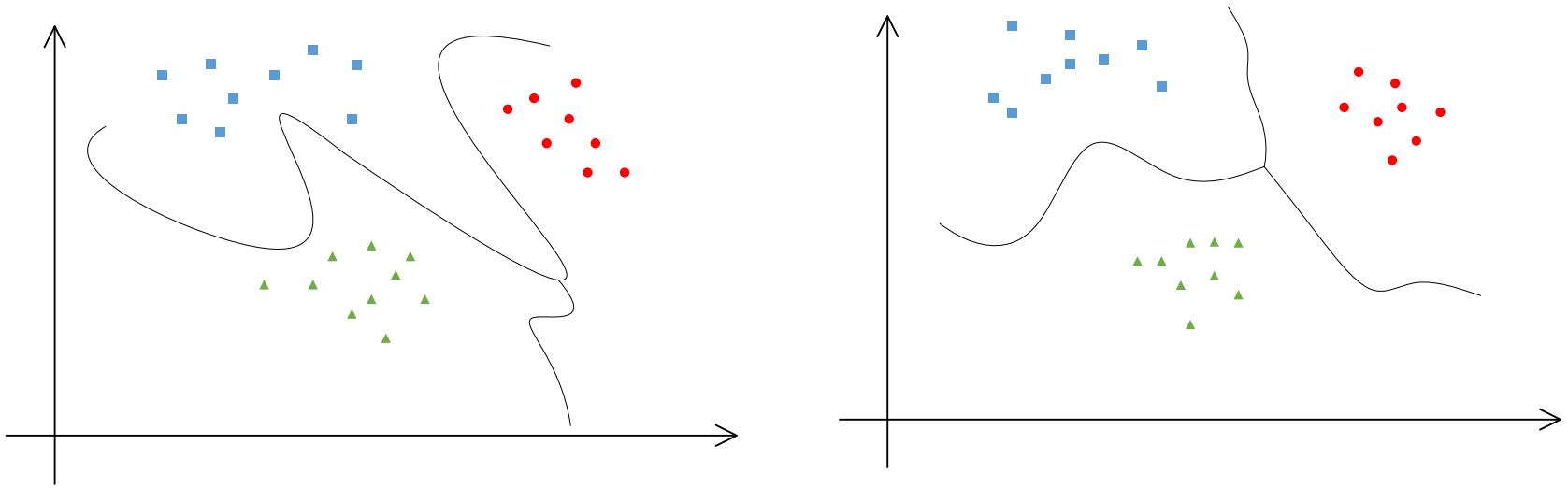
Defense at learning time

- *Data transformation:* filter/transform/process data to reduce the space of adversarial attacks [Dziugaite et al., 2016]



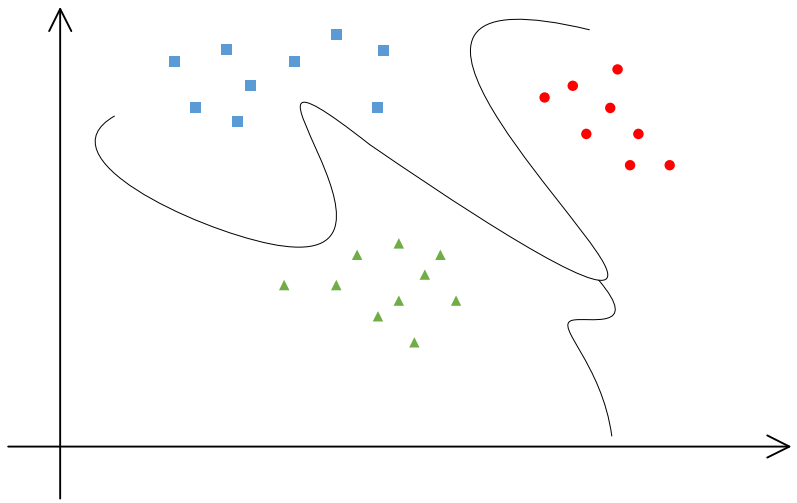
Defense at learning time

- *Distillation:* extract information (class probability) from the network to enhance its robustness [Papernot et al., 2016b]



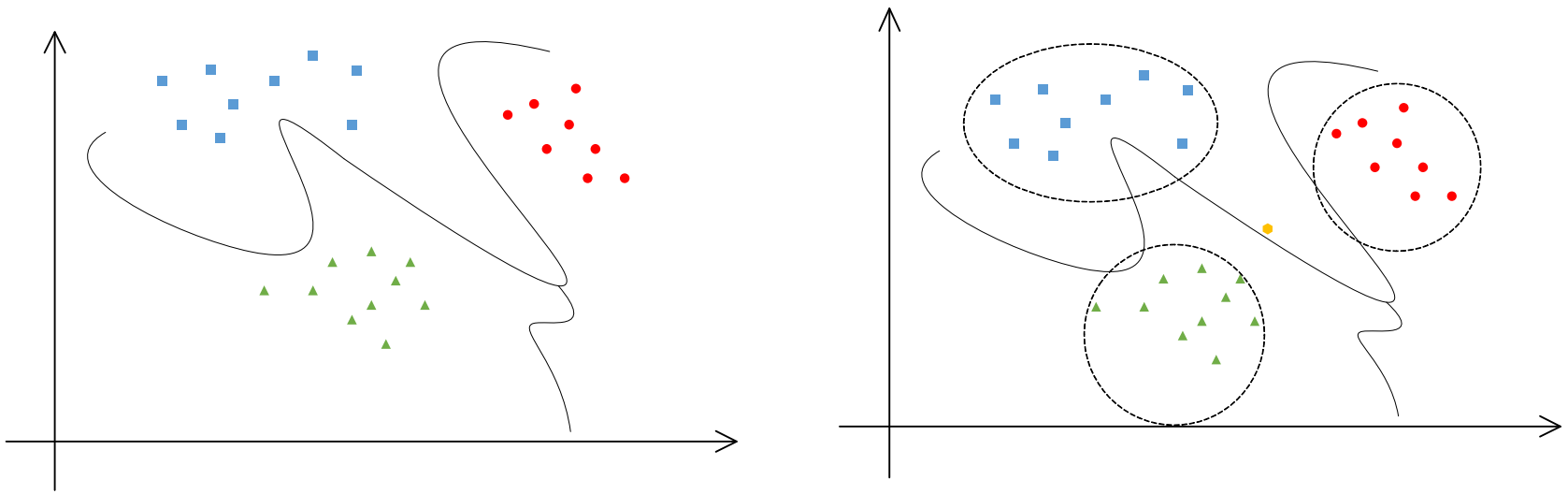
Defense at inference time

- *Gradient Masking:* penalize the degree of change in the output wrt the input. [Ross and Doshi-Velez, 2018]



Defense at inference time

- *Blind-spot evasion:* exclude points that do not behave like training samples [Melis et al., 2017]



Principles of Defense [Kolter and Madry, 2018; Biggio and Roli, 2018]

1. Do not train on untrusted data
2. Do not allow access to model to untrusted agents
3. Do not fully trust predictions

1. Design for security
2. Detect
3. Retrain
4. Verify

4. Safety of Machine Learning

AI Safety

Study of the broad impact of machine learning on the environment in which it is deployed.

- *Long-term AI safety:* concerned with existential risks [Bostrom, 2014]
- *AI Alignment:* aligning the goals of AI with the goals of the designers [Taylor et al., 2016]
- *Concrete AI safety:* current safety problem in machine learning [Amodei et al., 2016b]

Concrete AI Safety

- **Catastrophic Loss Function Misspecifications** [Amodei et al., 2016b]
  - *Incorrect formal loss function*
    - Negative side effects
    - Reward hacking
  - *Unlearnability of the loss function*
    - Scalable oversight
  - *Incorrect specification of the model*
    - Safe exploration
    - Robustness to distribution shift
- **Interpretability of the Learned Model**
- **Fairness of the Learned Model**

Other related topics: *ethics*; *privacy*; *policy*; *accountability*.

Avoiding Negative Side Effects

How do we guarantee that an agent will not cause bad side effects while pursuing its aim?

Example: If we train a cleaning robot whose loss function is proportional to the rubbish in a room, how do we guarantee it will not knock down furniture while cleaning up?

- Define or learn a reward function that penalizes changes to the environment
- Minimize *empowerment* of an agent [Salge et al., 2014]
- Combine different reward functions of multiple agents [Hadfield-Menell et al., 2016]
- Make reward function uncertain

Reward Hacking

How do we guarantee that an agent will not trick its loss function?

Example: If we train a cleaning robot whose loss function is proportional to the rubbish in a room, how do we guarantee it will not just disable its vision system?

- Adaptive or adversarial reward function
- Providing limited or blinded information about the environment
- Setting a cap on reward [Ajakan et al., 2014]
- Combine multiple reward functions [Deb, 2014]
- Instantiating trip wires

Scalable Oversight

How do we guarantee that an agent will learn every relevant aspect of its aim with a limited oversight?

Example: If we train a cleaning robot whose loss function is proportional to the rubbish in a room, how do we guarantee it will learn not to destroy valuable stray items on the floor?

- Train using aggregate or noisy information [Mann and McCallum, 2010]
- Hierarchical learning [Dayan and Hinton, 1993]

Safe Exploration

How do we guarantee that an agent will not undertake catastrophic actions while exploring?

Example: If we train a cleaning robot, how do we guarantee it will insert a wet mop into a plug?

- Use a risk-sensitive reward function accounting for worst-case scenario [Garcıa and Fernández, 2015]
- Learn from near-optimal demostrations [Abbeel and Ng, 2005]
- Train in a simulated environment
- Bound exploration
- Rely on human oversight [Saunders et al., 2017]

Robustness to Distribution Shift

How do we guarantee that an agent will behave consistently when the environment changes?

Example: If we train a cleaning robot in a house room, how do we guarantee it will behave safely in a factory?

- Rely on *covariate shift adaptation* [Sugiyama and Kawanabe, 2012]
- Devise algorithms to detect out-of-distribution conditions and devise appropriate strategies
- Increase and extend the training data [Amodei et al., 2016a]
- Model through counterfactual reasoning

Interpretability

How do we guarantee that decisions of machine learning systems can be explained and understood?

Example: If we use a machine learning model to decide on a loan, how do we guarantee the decision can be understood?

- Favour simple interpretable models [Lou et al., 2012; Caruana et al., 2015]
- Compress complex models
- Improve visualization techniques [Vellido et al., 2012]
- Use specific tools to get insights into complex models (e.g.: *saliency maps*) [Simonyan et al., 2013; Montavon et al., 2017]
- Interpret models locally [Ribeiro et al., 2016]

Fairness [Kusner et al., 2017]

How do we guarantee that decisions of machine learning systems do not create or spread biases?

Example: If we use a machine learning model to choose an employee, how do we guarantee it will not be affected by racial prejudices?

$$f : (X, A) \to Y$$

- Fairness through unawareness
- Individual fairness
- Demographic parity
- Equality of opportunity
- Counterfactual fairness [Pearl, 2009; Kusner et al., 2017]

Neglected aspects

- Role and robustness of *features*
- Vulnerabilities of *other forms of learning* (reinforcement learning)
- *Physical world* perturbations (3D printing)
- Security of *software*
- *Variety of perturbations* (e.g.: ℓ_p-norm, rotations)
- How robustness affects *decision boundaries*
- *Formal verifications* (Reluplex)
- *Game-theoretic approaches* (zero-sum games, Nash games)
- *Security-by-obscurity* (randomization)
- *Ensembling*
- ....

Thanks!

Thank you for listening!

References I

Pieter Abbeel and Andrew Y Ng. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 1–8. ACM, 2005.

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*, 2014.

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *arXiv preprint arXiv:1801.00553*, 2018.

References II

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International Conference on Machine Learning*, pages 173–182, 2016a.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016b.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016c.

References III

Giuseppe Ateniese, Luigi V Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali, and Giovanni Felici. Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. *International Journal of Security and Networks*, 10(3):137–150, 2015.

Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84: 317–331, 2018.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Support vector machines under adversarial label noise. In *Asian Conference on Machine Learning*, pages 97–112, 2011.

References IV

N. Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014. ISBN 9780199678112. URL `https://books.google.no/books?id=7_H8AwAAQBAJ`.

Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730. ACM, 2015.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

References V

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4): 303–314, 1989.

Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, Deepak Verma, et al. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108. ACM, 2004.

Peter Dayan and Geoffrey E Hinton. Feudal reinforcement learning. In *Advances in neural information processing systems*, pages 271–278, 1993.

Kalyanmoy Deb. Multi-objective optimization. In *Search methodologies*, pages 403–449. Springer, 2014.

References VI

Pedro Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87, 2012.

Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.

Luciano Floridi. *The philosophy of information*. Oxford University Press, 2011.

Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *USENIX Security Symposium*, pages 17–32, 2014.

References VII

Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

Dylan Hadfield-Menell, Stuart J Russell, Pieter Abbeel, and Anca Dragan. Cooperative inverse reinforcement learning. In *Advances in neural information processing systems*, pages 3909–3917, 2016.

References VIII

Marius Kloft and Pavel Laskov. Online anomaly detection under adversarial impact. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 405–412, 2010.

Zico Kolter and Aleksander Madry. Adversarial robustness: Theory and practice. 12 2018.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pages 4069–4079, 2017.

References IX

Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 150–158. ACM, 2012.

David J.C. MacKay. *Information theory, inference, and learning algorithms*, volume 7. Cambridge University Press, 2003.

Gideon S Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *Journal of machine learning research*, 11(Feb):955–984, 2010.

Shike Mei and Xiaojin Zhu. Using machine teaching to identify optimal training-set attacks on machine learners. In *AAAI*, pages 2871–2877, 2015.

References X

Marco Melis, Ambra Demontis, Battista Biggio, Gavin Brown, Giorgio Fumera, and Fabio Roli. Is deep learning safe for robot vision? adversarial examples against the icub humanoid. In *Computer Vision Workshop (ICCVW), 2017 IEEE International Conference on*, pages 751–759. IEEE, 2017.

Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 2017.

Mehran Mozaffari-Kermani, Susmita Sur-Kolay, Anand Raghunathan, and Niraj K Jha. Systematic poisoning attacks on and defenses for machine learning in healthcare. *IEEE journal of biomedical and health informatics*, 19(6):1893–1905, 2015.

References XI

Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. Towards the science of security and privacy in machine learning. *arXiv preprint arXiv:1611.03814*, 2016a.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016b.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Roberto Perdisci, David Dagon, Wenke Lee, Prahlad Fogla, and Monirul Sharif. Misleading worm signature generators using deliberate noise injection. In *Security and Privacy, 2006 IEEE Symposium on*, pages 15–pp. IEEE, 2006.

References XII

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Christoph Salge, Cornelius Glackin, and Daniel Polani. Empowerment–an introduction. In *Guided Self-Organization: Inception*, pages 67–114. Springer, 2014.

References XIII

William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Dawn Song. Ai and security: Lessons, challenges and future directions. 07 2018.

Jacob Steinhardt, Pang Wei Koh, and Percy Liang. Certified defenses for data poisoning attacks. *arXiv preprint arXiv:1706.03691*, 2017.

References XIV

Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: introduction to covariate shift adaptation*. MIT Press, 2012.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Jessica Taylor, Eliezer Yudkowsky, Patrick LaVictoire, and Andrew Critch. Alignment for advanced machine learning systems. *Machine Intelligence Research Institute*, 2016.

References XV

Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *USENIX Security Symposium*, pages 601–618, 2016.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. There is no free lunch in adversarial robustness (but there are unexpected benefits). *arXiv preprint arXiv:1805.12152*, 2018.

Alfredo Vellido, José David Martín-Guerrero, and Paulo JG Lisboa. Making machine learning models interpretable. In *ESANN*, volume 12, pages 163–172. Citeseer, 2012.

References XVI

David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *Evolutionary Computation, IEEE Transactions on*, 1(1):67–82, 1997.