Elements of Adversarial Machine Learning

Dimitris Diochnos

School of Computer Science
University of Oklahoma

October 18, 2020
Norman, OK

Outline

Outline

What is Machine Learning?

- Learning from historical data to make decisions about unseen data.

- Traditional Programming

  Data $\longrightarrow$ $\boxed{\text{Computer}}$ $\longrightarrow$ Output
  Program $\longrightarrow$

- Machine Learning

  Data $\longrightarrow$ $\boxed{\text{Computer}}$ $\longrightarrow$ Program
  Output $\longrightarrow$

Machine Learning: A Success Story

Machine learning (ML) has changed our lives.

- Health
- Finance/Economy
- Computer vision: autonomous driving
- Computer security: threat prediction
- many more applications ...

Machine Learning in the Presence of Adversaries

- Machine learning was not designed to deal with adversaries.
  - 'Naive' requirement for success: make few mistakes on average.

Machine Learning in the Presence of Adversaries

- Machine learning was not designed to deal with adversaries.
  - 'Naive' requirement for success: make few mistakes on average.

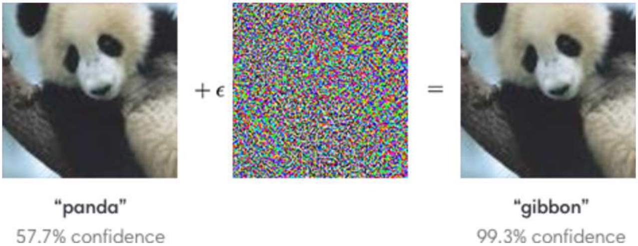
What is the performance of ML systems
in the presence of (malicious) adversaries ?

Machine Learning in the Presence of Adversaries

- Machine learning was not designed to deal with adversaries.
  - 'Naive' requirement for success: make few mistakes on average.

What is the performance of ML systems
in the presence of (malicious) adversaries ?

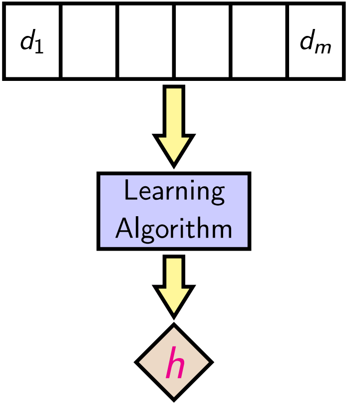
- Subverting spam filter by poisoning training data [Nelson et. al. 2008]
- Evading PDF malware detectors [Xu et. al. 2016]
- Fooling computer vision systems by adding small perturbations [Szegedy et. al. 2014]



Classification

Training

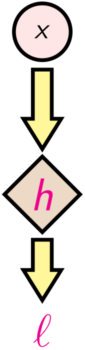
$$x_i \sim D$$
$$d_i = (x_i, c(x_i))$$



$$Conf(L) = \Pr\left(\mathbf{Risk}_D(h, c) < \varepsilon\right)$$

Testing

$$x \sim D$$
$$d = (x, c(x))$$



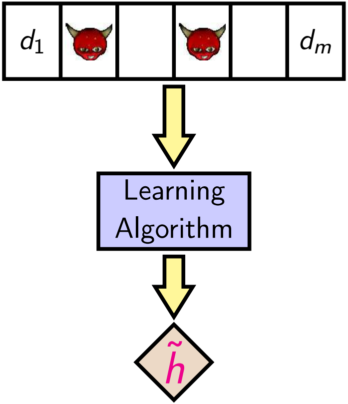
$$\mathbf{Risk}_D(h, c) = \Pr_D(\ell \neq c(x))$$

Classification under Attack

Poisoning Attack

$x_i \sim D$

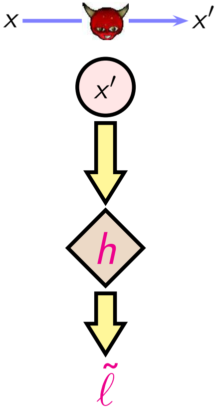
$d_i = (x_i, c(x_i))$



Evasion Attack

$x \sim D$

$d = (x, c(x))$



Terminology and Goal of Learning

Goal (Good Approximation with High Probability)
There is a function c over a space X. One wants to come up (in a reasonable amount of time) with a function h such that h is a *good approximation* of c with *high probability*.

Description 1 (Parameters and Terminology)

- X: *Instance Space* *(say, $\{0,1\}^n$)*
- $\mathcal{Y}$: *Labels* *(say, $\{+,-\}$)*
- $c \in \mathcal{C}$: *Target concept belonging to a concept class*
- $h \in \mathcal{H}$: *Hypothesis belonging to a hypothesis class*
- *Good Approximation: Small Risk (Error)* ε
- *High Probability: Confidence* $1-\delta$
- *Reasonable Amount of Time: Polynomial in* $n, 1/\varepsilon, 1/\delta$

Important Questions in Adversarial Machine Learning

- Formalizing (complexity-theoretic) notions of security.
- What are the inherent powers and limitations of adversaries against ML systems?
- Barriers for provable robustness of ML systems against adversarial attacks, whether poisoning or evasion.
  - information-theoretic, with all-knowing adversaries
  - computationally bounded adversaries
- Can ML systems achieve Probably Approximately Correct (PAC) generalization bounds under adversarial attacks?

Important Questions in Adversarial Machine Learning

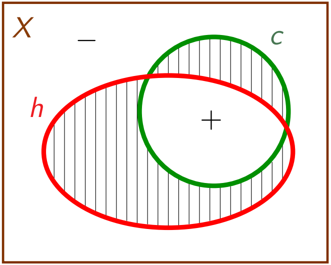
- Formalizing (complexity-theoretic) notions of security.
  [New definition and comparative study]
- What are the inherent powers and limitations of adversaries against ML systems?
  [Concentration of measure]
- Barriers for provable robustness of ML systems against adversarial attacks, whether poisoning or evasion.
  - information-theoretic, with all-knowing adversaries
  - computationally bounded adversaries

  [Concentration of measure]
- Can ML systems achieve Probably Approximately Correct (PAC) generalization bounds under adversarial attacks?
  [PAC learning under poisoning; positive & negative results]

Outline

Probably Approximately Correct (PAC) Learning

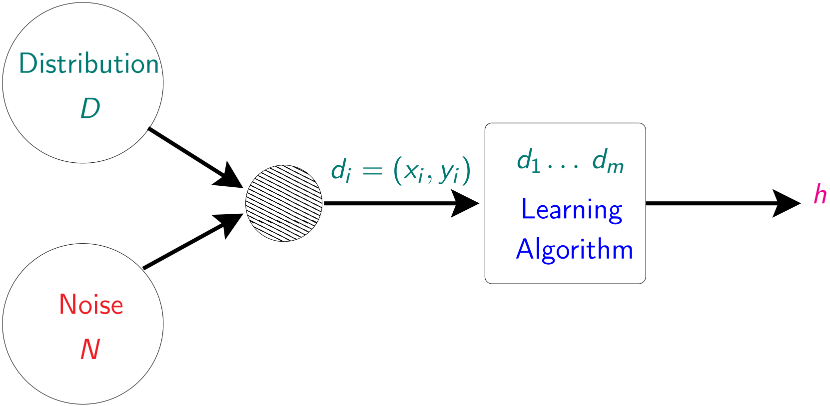
- There is an *arbitrary, unknown* distribution $\mathcal{D}$ over X.
- Learn from *poly* $\left(\frac{1}{\varepsilon}, \frac{1}{\delta}\right)$ many examples $(x, \mathrm{c}(x))$, where $x \sim \mathcal{D}$.
- $\mathrm{Risk}_{\mathcal{D}}(\mathrm{h}, \mathrm{c}) = \mathrm{Pr}_{x\sim\mathcal{D}}(\mathrm{h}(x) \neq \mathrm{c}(x))$.



Goal 1 ([Valiant, 1984])

$$\Pr\left(Risk_{\mathcal{D}}(h, c) \leq \varepsilon\right) \geq 1 - \delta\ .$$

PAC Learning under Noise

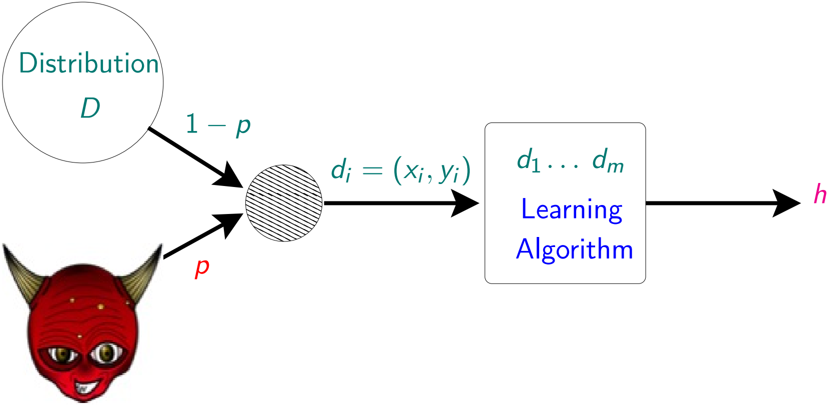


- Goal

$$\Pr\left(\mathrm{Risk}_{\mathcal{D}}(\mathrm{h}, \mathrm{c}) \leq \varepsilon\right) \geq 1 - \delta$$

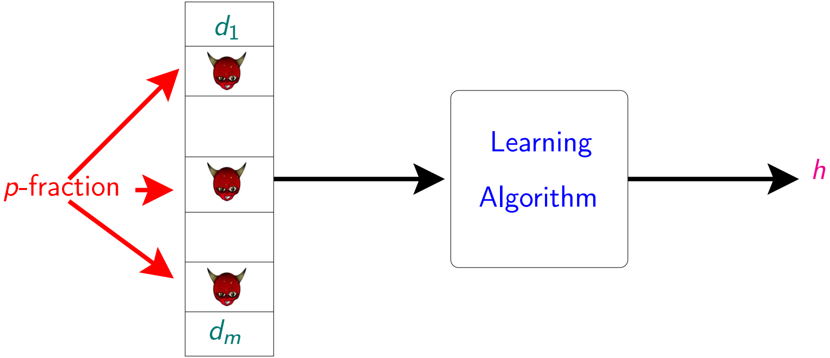
- *poly* $\left(\frac{1}{\varepsilon}, \frac{1}{\delta}\right)$ many examples

Malicious Noise Model [Valiant, 1985]



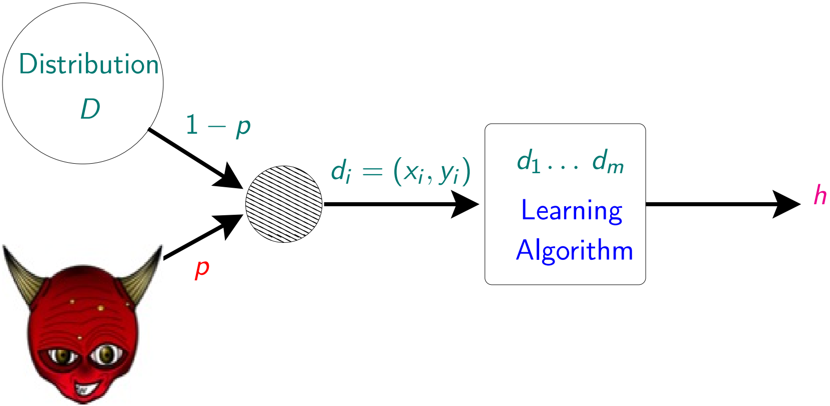
- Adversary may use arbitrary (x_i, y_i)
- e.g., wrong label $((x_i, y_i) \notin Supp(\mathcal{D}))$

Poisoning Attacks



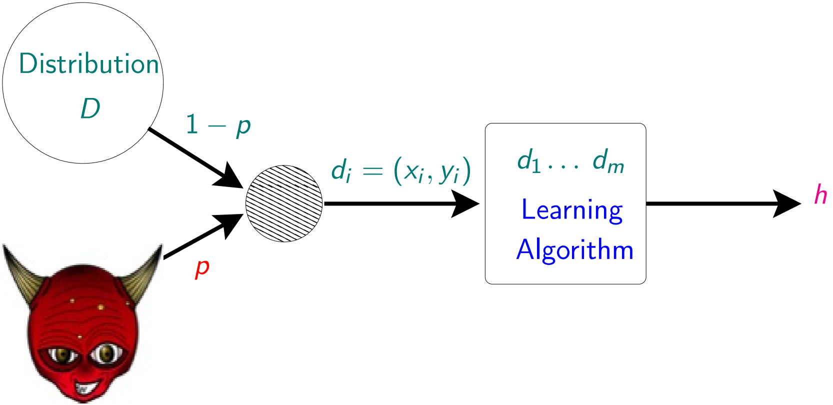
- Adversary knows the test example (targeted)
- Adversary does not know the test example (non-targeted)
- [Xiao, Biggio, Brown, Fumera, Eckert, Roli, 2015]
- [Shen, Tople, Saxena, 2016]
- ...

Is PAC Learning Possible under Malicious Noise?



- PAC learning not possible under malicious noise [Kearns & Li, 1993]
  - Using wrong labels
  - Using specific pathological distribution
    (method of induced distributions)

Limiting the Power of the Adversary under Malicious Noise



- What if the adversary can not give wrong labels?
- What if we care about specific (natural) distributions?
- Is PAC learning possible now?

p-Tampering Noise/Attack Model

- Each training example

$(x_i, y_i) \sim \mathcal{D}$, with probability $1 - p$

$(x_i, y_i) \sim$  , with probability p

-  knows the history of examples so far
-  can only generate outputs from $\text{Supp}(\mathcal{D})$
  - i.e., adversary always uses correct label y_i
- [Per Austrin, Kai-Min Chung, Mohammad Mahmoody, Rafael Pass & Karn Seth, 2014]
- [Mahloujifar & Mahmoody, 2017]
- [Mahloujifar, Diochnos & Mahmoody, 2018]
- Defensible malicious noise

Main Questions [Mahloujifar, D, Mahmoody, ALT 2018]

1. Qualitative: Is PAC learning possible under p-tampering attacks? (when it is possible under no attacks)

2. Quantitative: How much can a p-tampering attack increase the risk?

Main Questions [Mahloujifar, D, Mahmoody, ALT 2018]

1. Qualitative: Is PAC learning possible under p-tampering attacks? (when it is possible under no attacks)
   Answer:

   YES

2. Quantitative: How much can a p-tampering attack increase the risk?
   Answer: For 'bounded' loss functions, non-targeted case,

$$\mathrm{Risk}_{\mathcal{D}}(h) \longrightarrow \mathrm{Risk}_{\mathcal{D}}(h) + p \cdot \mathrm{Var}[\mathrm{Risk}_{\mathcal{D}}(h)]$$

$$\Pr(\mathrm{Risk}_{\mathcal{D}}(h) \geq \varepsilon) = \delta \longrightarrow \Pr(\mathrm{Risk}_{\mathcal{D}}(h) \geq \varepsilon) \geq \delta + p\delta(1-\delta)$$

Positive Result: Feasibility of PAC Learning

- Is PAC learning possible under p-tampering attacks?
  (when it is possible under no attacks)

Yes

Positive Result: Feasibility of PAC Learning

Theorem 1 (Informal)

PAC learning a concept class $\mathcal{C}$ under no noise

$$\Longrightarrow$$

PAC learning $\mathcal{C}$ under p-tampering attacks

Positive Result: Feasibility of PAC Learning

Theorem 1 (Informal)

PAC learning a concept class $\mathcal{C}$ under no noise

$$\Longrightarrow$$

PAC learning $\mathcal{C}$ under p-tampering attacks

Proof Sketch

- With probability p the adversary can change each training example.
- About $(1 - p)$ fraction of the data is generated honestly.
- Require $m' \approx \frac{m}{1-p}$ examples in this adversarial setting.
  (m examples enough for PAC learning without noise)

Positive Result: Feasibility of PAC Learning

Theorem 1 (Informal)

PAC learning a concept class $\mathcal{C}$ under no noise

$$\Longrightarrow$$

PAC learning $\mathcal{C}$ under p-tampering attacks

Proof Sketch

- With probability p the adversary can change each training example.
- About $(1-p)$ fraction of the data is generated honestly.
- Require $m' \approx \frac{m}{1-p}$ examples in this adversarial setting.
  (m examples enough for PAC learning without noise)

Remark 1

The locations of the examples that are replaced are outside of the adversary's control.

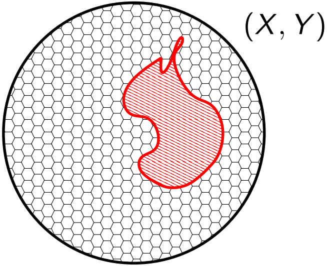
Positive Result: Feasibility of PAC Learning

Theorem 1 (Informal)

PAC learning a concept class $\mathcal{C}$ under no noise

$$\Longrightarrow$$

PAC learning $\mathcal{C}$ under p-tampering attacks



- Theorem no longer holds if the adversary can choose the location
- e.g., learner never sees examples from the shaded region.

Random vs Adversarial Locations

- *p*-Tampering vs Bounded Budget

p-Tampering: The adversary can not choose which examples to alter.

$$\begin{cases} (x_i, y_i) \sim \mathcal{D} & , \quad \text{with probability } 1 - p \\ (x_i, y_i) \sim & , \quad \text{with probability } p \end{cases}$$

Bounded Budget: The adversary can choose which *p*-fraction of the training examples to alter.

- Query learning; [Angluin, Kriķis, Sloan, Turán, 1997]
- Strong adaptive corruption; [Goldwasser, Kalai, Park, 2015]
- The previous theorem does not extend to the bounded budget case.

Main Questions [Mahloujifar, D, Mahmoody, ALT 2018]

1. Qualitative: Is PAC learning possible under p-tampering attacks? (when it is possible under no attacks)
   Answer:

   YES ☑

2. Quantitative: How much can a p-tampering attack increase the risk?
   Answer: For 'bounded' loss functions, non-targeted case,

$$\mathrm{Risk}_{\mathcal{D}}(h) \longrightarrow \mathrm{Risk}_{\mathcal{D}}(h) + p \cdot \mathrm{Var}[\mathrm{Risk}_{\mathcal{D}}(h)]$$

$$\Pr(\mathrm{Risk}_{\mathcal{D}}(h) \geq \varepsilon) = \delta \longrightarrow \Pr(\mathrm{Risk}_{\mathcal{D}}(h) \geq \varepsilon) \geq \delta + p\delta(1-\delta)$$

Idea for Answering the Second Question in One Slide

- Attack designed to generate a specific joint distribution

$$\Pr(d_1, \ldots, d_m) = \Pr_{\mathcal{D}^m}(d_1, \ldots, d_m)\left(1 + p\left(f(d_1, \ldots, d_m) - \mathrm{E}_{\mathcal{D}^m}[f]\right)\right) .$$

- Expected value under new distribution is,

$$\mathrm{E}[f] \geq \mathrm{E}[f] + p \cdot \mathrm{Var}[f]$$

- Generalized Santha-Vazirani source [Santha & Vazirani, 1986], [Beigi, Etesami, Gohari, 2017]
    - generated by an efficient p-tampering attack

Forming a Better Picture on Poisoning Attacks

- These were polynomial-time attacks and defenses.

- What are the ultimate powers of adversaries on poisoning attacks – without even taking computational complexity into account?

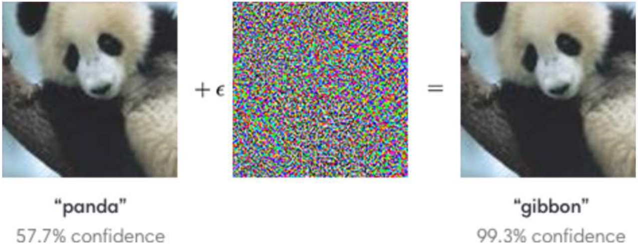
Forming a Better Picture on Poisoning Attacks

- These were polynomial-time attacks and defenses.

- What are the ultimate powers of adversaries on poisoning attacks – without even taking computational complexity into account?
  - Connection with the phenomenon of concentration of measure.
    - We will see attacks that are stronger (smaller perturbations)
    - We will see attacks that are weaker (information-theoretic)
  - First we need to detour to adversarial examples, use notions from results there, and eventually connect such results to poisoning attacks as well.

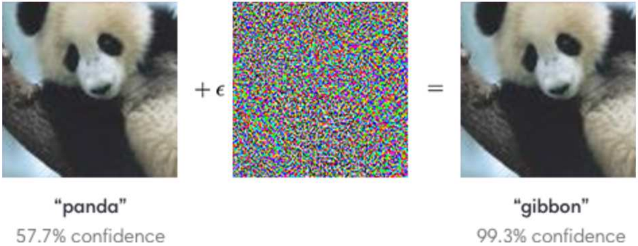
Outline

Adversarial Examples



- prediction change [Moosavi-Dezfooli et al., 2016], [Goodfellow et al., 2018], ...
- corrupted instance [Madry et al., 2018], [Wong & Kolter, 2018], ... (earlier in different context; [Mansour et al., 2015], [Feige et al., 2015], ...)
- error region [Diochnos et al., 2018] (around the same time [Gilmer et al., 2018], [Bubeck et al., 2018], and more people are following; e.g., [Degwekar & Vaikuntanatan, 2019])

Adversarial Examples

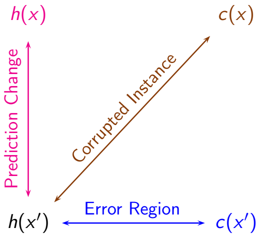


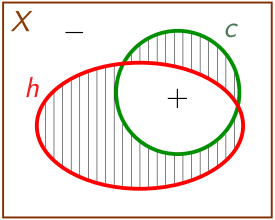
- prediction change [Moosavi-Dezfooli et al., 2016], [Goodfellow et al., 2018], ...
- corrupted instance [Madry et al., 2018], [Wong & Kolter, 2018], ...
  (earlier in different context; [Mansour et al., 2015], [Feige et al., 2015], ...)
- error region [Diochnos et al., 2018]
  (around the same time [Gilmer et al., 2018], [Bubeck et al., 2018], and more people are following; e.g., [Degwekar & Vaikuntanatan, 2019])
- Definitions coincide in the case of images.
- Definitions diverge in other natural cases. [Diochnos et al., 2018]

Related Work on Certified Robustness

- Cross-Lipschitz regularization [Hein & Andriushchenko, 2017]
- Earth-mover's distance between distributions [Sinha et al., 2018]
- Semidefinite relaxation [Raghunathan et al., 2018]
- Convex / linear programming relaxation [Wong & Kolter, 2018], [Wong et al., 2018]
- Connections to robust optimization [Ben Tal et al., 2009]
- Ultimately want provable guarantees, better results and understanding.
  - Understand robustness beyond image classification.
  - Hard to interpret results of corrupted instances in some natural contexts (e.g., uniform distribution over $\{0,1\}^n$)
  - Guarantee misclassification (adversarial examples) with error-region definition.

Understanding the Different Definitions





- All three definitions **coincide for images**
  - truth proximity assumption (corrupted instance, prediction change)
  - initial correctness assumption (prediction change)
- Only error-region guarantees misclassification!

Formalizing Adversarial Risk and Adversarial Robustness

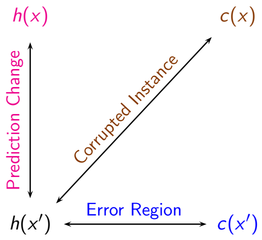
- $\mathcal{B}all_r(x) = \{x' \in X \mid \mathrm{d}(x, x') \leq r\}$ (e.g., d is HD over $\{0,1\}^n$)

Definition 2 (Error-Region Adversarial Risk)

$$\mathrm{Risk}_r^{\mathrm{ER}}(h, c) = \Pr_{x \leftarrow D}[\exists x' \in \mathcal{B}all_r(x), h(x') \neq c(x')].$$

Definition 3 (Error-Region Adversarial Robustness)

$$\mathrm{Rob}^{\mathrm{ER}}(h, c) = \mathrm{E}_{x \leftarrow D}\left[\inf\{r \colon \exists x' \in \mathcal{B}all_r(x), h(x') \neq c(x')\}\right].$$



Main Questions [D, Mahloujifar, Mahmoody, NeurIPS 2018] and [Mahloujifar, D, Mahmoody, AAAI 2019]

1. Does it matter which definition we use for adversarial examples? (if we want to **guarantee misclassification**)

2. Are there inherent reasons enabling evasion attacks?

Main Questions [D, Mahloujifar, Mahmoody, NeurIPS 2018] and [Mahloujifar, D, Mahmoody, AAAI 2019]

1. Does it matter which definition we use for adversarial examples? (if we want to **guarantee misclassification**)
   Answer:

   YES

   (PC/CI may imply **incorrect certified robustness** compared to ER)

2. Are there inherent reasons enabling evasion attacks?
   Answer:

   Concentration of measure

   (actually the analysis also applies to poisoning attacks)

Incorrect Definitions May Lead to Catastrophe

Couplas in Finance

- Formula to compute risk in correlated assets, by David X. Li (2000)
- **Story:** *Recipe for disaster: the formula that killed Wall Street*, in the `Wired` magazine. (https://www.wired.com/2009/02/wp-quant/)
- **Talk:** *On Models & Theory*, by Elchanan Mossel (v=mg2k1dwByn8)

  "... many practitioners use mathematics or methods that they do not understand and this often leads to disastrous results and I think the collapse in Wall Street is one of them!"

  — Elchanan Mossel, 2016

- We will study monotone conjunctions under the uniform distribution to prove large discrepancies on the robustness predicted by the error region definition and the other two definitions.

Why Monotone Conjunctions? Why Uniform Distribution?

- What are these functions?
  - Logical AND of a subset of the variables $\{x_1, \ldots, x_n\}$.
  - Say $n \geq 5$. Then, for example, $c = x_2 \wedge x_4 \wedge x_5$.
- One of the most basic ways of selecting (combining) features (constraints) in a prediction mechanism.
- Building block for other classes of functions that are less understood; e.g., monotone DNF formulae.
- Typical benchmark (together with halfspaces and general conjunctions) for studying various concepts in learning theory as it usually provides interesting, but non-trivial insights, of the definitions, the bounds that we should expect to get, etc.
- Uniform distribution U_n is perhaps the most natural distribution to think of and the *de-facto* benchmark on any problem that we want to understand better.

Finding All Common Properties of a Set of Objects

Let $X = \{0, 1\}^8$ and $c = x_2 \wedge x_4 \wedge x_5$.

- Request m examples and look at the positive ones.
- Delete the variables that are falsified by the positive examples.

A Study of Thinking [Bruner, Goodnow, Austin, 1956]

Finding All Common Properties of a Set of Objects

Let $X = \{0, 1\}^8$ and $c = x_2 \wedge x_4 \wedge x_5$.

- Request m examples and look at the positive ones.
- Delete the variables that are falsified by the positive examples.

A Study of Thinking [Bruner, Goodnow, Austin, 1956]

example	hypothesis h
	$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_7 \wedge x_8$
$((11011101), +)$	$x_1 \wedge x_2 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_8$
$((01011111), +)$	$x_2 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_8$
$((01011100), +)$	$x_2 \wedge x_4 \wedge x_5 \wedge x_6$

Finding All Common Properties of a Set of Objects

Let $X = \{0,1\}^8$ and $c = x_2 \wedge x_4 \wedge x_5$.

- Request m examples and look at the positive ones.
- Delete the variables that are falsified by the positive examples.

A Study of Thinking [Bruner, Goodnow, Austin, 1956]

example	hypothesis h
	$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_7 \wedge x_8$
$((11011101), +)$	$x_1 \wedge x_2 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_8$
$((01011111), +)$	$x_2 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_8$
$((01011100), +)$	$x_2 \wedge x_4 \wedge x_5 \wedge x_6$

- Is such an algorithm good for PAC learning?
  - YES, provided m is large enough.
  - Creates a consistent hypothesis:
    - Predicts correct label for each training example.

Case Study: Monotone Conjunctions under U_n

- $\mathcal{H} = \mathcal{C} =$ monotone conjunctions having at least one and at most n Boolean variables.
- $|h| =$ number of variables in h $\qquad (h_1 = x_1 \wedge x_5 \wedge x_8 \Rightarrow |h_1| = 3)$

$$\mathrm{c} = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{k=1}^{u} y_k \qquad \text{and} \qquad \mathrm{h} = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{\ell=1}^{w} z_\ell\,. \tag{1}$$

$$\mathcal{E}\left(\mathrm{h}, \mathrm{c}\right) = \{x \in \{0,1\}^n \mid \mathrm{h}(x) \neq \mathrm{c}(x)\}\,.$$

$$\Pr_{x \leftarrow U_n}\left[x \in \mathcal{E}\left(h, c\right)\right] = 2^{-|c|} + 2^{-|h|} - 2^{1-m-u-w}\,.$$

-  has oracle access to $h \Longrightarrow$  efficiently reconstructs h.
  - For $i \in \{1, \ldots, n\}$ query $x_i = \langle 1, \ldots, 1, 0, 1, \ldots, 1\rangle$
    ($x_{one} = \langle 1, \ldots, 1\rangle$ is always $+$)

Case Study: Monotone Conjunctions under U_n

Theorem 4 (Error Region Robustness; [D, Mahloujifar, Mahmoody, NeurIPS2018])

- *If* $h = c$, *then* $\mathrm{Rob}^{\mathrm{ER}}(h, c) = \infty$
- *If* $h \neq c$, *then* $\frac{1}{16} \cdot \min\{|h|, |c|\} \leq \mathrm{Rob}^{\mathrm{ER}}(h, c) \leq 1 + \min\{|h|, |c|\}$.

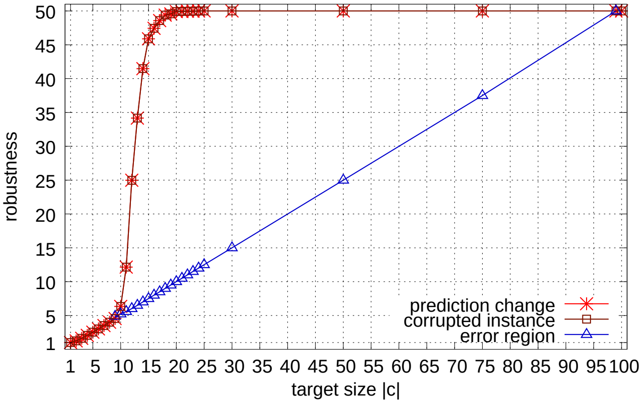
Theorem 5 (Prediction Change Robustness; [D, Mahloujifar, Mahmoody, NeurIPS2018])

$\mathrm{Rob}_r^{\mathrm{PC}}(h) = |h|/2 + 2^{-|h|}$.

Theorem 6 (Corrupted Instance Robustness; [D, Mahloujifar, Mahmoody, NeurIPS2018])

$|h|/4 < \mathrm{Rob}^{\mathrm{CI}}(h, c) < |h| + 1/2$.

Evading Monotone Conjunctions under U_n



- $n = 100$, $\varepsilon = 0.01$, $\delta = 0.05 \Rightarrow m = \left\lceil \frac{1}{\varepsilon} \cdot \ln\left(\frac{|\mathcal{H}|}{\delta}\right) \right\rceil = 7,232$ examples
- For each $|c|$ perform 500 runs,
  - estimate robustness using 10K examples each time.

Main Questions [D, Mahloujifar, Mahmoody, NeurIPS 2018] and [Mahloujifar, D, Mahmoody, AAAI 2019]

1. Does it matter which definition we use for adversarial examples? (if we want to **guarantee misclassification**)
   Answer:

   YES ☑

   (PC/CI may give **wrong certified robustness** compared to ER)

2. Are there inherent reasons enabling evasion attacks?
   Answer:

   Concentration of measure

   (actually the analysis also applies to poisoning attacks)

Why Concentration of Measure?

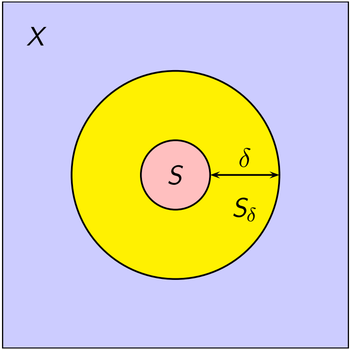
- Because making small changes on any given instance (say w.r.t. HD over $\{0,1\}^n$) allows us to generate clouds of neighboring points that have cummulatively higher probability mass.
- So, with such small changes we can cover quickly almost the entire space (say 99%).

Concentration of Measure

Definition 7 (δ-expansion)

The δ-expansion of $S \subseteq X$ is: $S_\delta = \{x \in X \mid d(x, S) \leq \delta\}$

- $\Pr_D(S) = 1/2 \Rightarrow \Pr_D(S_\delta) \to 1$ exponentially quickly as $\delta \nearrow$
  $\Rightarrow \Pr(S_\delta) \approx 1$ for $\delta \ll diam_d(X)$.



Examples of Concentrated Spaces

Normal Lévy families

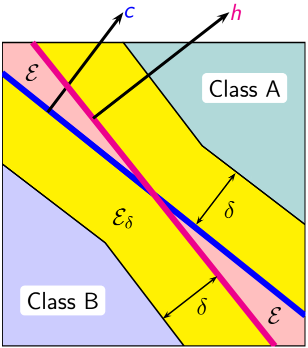
- For any set S such that $\Pr(S) = 1/2$ and $\delta \approx 1/\sqrt{n}$ we have $\Pr(S_\delta) \geq 0.99$.

Examples of Normal Lévy families

- n-dimensional Gaussian with $d = \ell_2$.
- Product distribution over $\{0,1\}^n$ with $d = HD$

Implication to Evasion Attacks

- Error region:
  - $\mathcal{E} = \{x \in X \mid h(x) \neq c(x)\}$.
- Adversarial risk:
  - $\mathrm{Risk}_{D,\delta}(\mathrm{h}, \mathrm{c}) = \mathrm{Pr}_D(\mathcal{E}_\delta)$.



Theorem 8 (Adversarial Examples for Normal Lévy Families)

Let (D, d) be a Lévy family with dimension n and diameter 1. Let h be a hypothesis such that $\mathrm{Risk}_D(h, c) \geq 1/poly(n)$. Then,  *with budget $\delta = \widetilde{\mathcal{O}}(1/\sqrt{n})$ can drive the $\mathrm{Risk}_{D,\delta}(h, c) \approx 1$.*

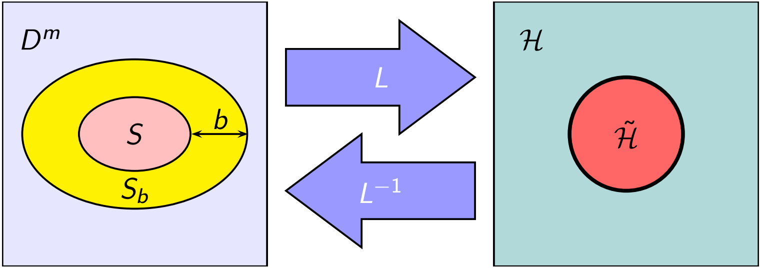
Implication to Poisoning Attacks

- Learner L uses a sample $S \sim D^m$.
- Let $\tilde{\mathcal{H}} \subseteq \mathcal{H}$ be the set of bad hypotheses (e.g., large risk)

Confidence: $Conf(L) = \Pr_{S \sim D^m}\left(L(S) \in \mathcal{H} \setminus \tilde{\mathcal{H}}\right)$

Adversarial Confidence:

$$Conf_b(L) = \Pr_{S \sim D^m}\left((\forall S')(d(S, S') \leq b) \mid L(S') \in \mathcal{H} \setminus \tilde{\mathcal{H}}\right)$$



Poisoning Attacks from Concentration

Theorem 9

Let L be a learner and $\tilde{\mathcal{H}}$ a subset of $\mathcal{H}$ where for each $h \in \tilde{\mathcal{H}}$ we have $Risk_D(h, c) > 1/\operatorname{poly}(m)$. Then,  *with budget $b = \widetilde{\mathcal{O}}\left(\sqrt{m}\right)$ can $\Pr\left(h \in \tilde{\mathcal{H}}\right) \approx 1$ ($Conf_b(L) \approx 0$) while the poisoned data are all still* ***correctly labeled****!*

Outline

Summary

- PAC learning is possible under poisoning attacks:
  - p-tampering with clean labels
  - weak p-budget with clean labels
- PAC learning is not possible under strong p-budget poisoning attacks.
- p-tampering can increase the risk by an amount of $p \cdot \mathrm{Var}[\mathrm{Risk}_{\mathcal{D}}(h, c)]$.
- Error-region guarantees misclassification of adversarial examples.
  - Other definitions may lead to incorrect bounds.
- Concentration of measure implies that adversarial examples almost always exist with an $\mathcal{O}\left(\sqrt{n}\right)$ perturbation.
- Substituting $\mathcal{O}\left(\sqrt{m}\right)$ training examples allows an adversary to almost always lead the learner towards a bad hypothesis.