Robust Machine Learning in the Adversarial Settings

Han Xiao, Huang Xiao

IT Security, Informatics (I20)
Technical University of Munich
xiaoh@in.tum.de

May 3, 2012
Overview

1. Examples
2. Philosophy
3. Adversarial Learning
4. Preliminaries
5. Exploratory Attack
6. Causative Attack
7. Application
8. Summary
Spam Disguise

Adding noise to the junk mail

Nnus ¤ yuang.fong@msa.hinet.net

to wzg0770

Why is this message in Spam? We've found that lots of messages from yuang.fong@msa.hinet.

tyx 您 nip 好： qdk

mzx 附 k 件 cp 是 vi 课 lqw 程 m 详 vip 细 hli 内 dq 容，rv 请 up 您 anc 参 u 阅！ je0KWT5u

jygt ¤ kkq@suijian.com

to me

Why is this message in Spam? It's similar to messages that

ポゆちかゅャケ 売様 厣

企业白领核心办公技能(PPT+Excel)企业高级应用2012
Spam Disguise

How?

- Create dummy@gmail.com
- Generate disguised spams and sends to dummy@gmail.com
- Select the most desired modification from the inbox.
Spam Disguise

How?

- Create dummy@gmail.com
- Generate disguised spams and sends to dummy@gmail.com
- Select the most desired modification from the inbox.

Questions:

- How to generate efficiently?
- What is “most desired”?
Crowd-Sourcing Data
Labelers may have different levels of expertise
Crowd-Sourcing Data
Labelers may have different levels of expertise
Crowd-Sourcing Data

Labelers may have different levels of expertise

What if haters dominates? Are they going to subvert the learning algorithm? How to recover the unbiased labels/ratings?
Why study adversarial learning?

If you know your enemies and know yourself, you will not be imperiled in a hundred battles; if you do not know your enemies but do know yourself, you will win one and lose one; if you do not know your enemies nor yourself, you will be imperiled in every single battle.

–Sun Tzu, *The Art of War*, 544 BC
Research Points

Optimal Attack Strategies
- Exploratory Attack
- Causative Attack
- Reverse-engineering

Robust Learning Algorithms
- Learning from Crowds
- Robust Active/Online Learning
An Endless War between Adversary and Defender

Han Feizi, *A Critique of the Doctrine of Position*, 256 BC

Figure: “How about use your spear to attack your shield?”
Related Knowledge

- Machine learning
- Convex geometry
- Optimization
- Game theory
Exploratory Attack and Causative Attack

Adversarial settings

The adversary will manipulate instances to mislead the decision of the classifier in their favor.

*Exploratory attack*

- on test phrase;
- disguise a malicious instance to evade from being detected.

*Causative attack*

- on training phrase;
- manipulate the training set to subvert the learning process.
Classification Algorithm

Notations (on binary classification)

Input space: $\mathcal{X} \subseteq \mathbb{R}^D$

Response space: $\mathcal{Y} := \{-1, 1\}$

Instance: $x \in \mathcal{X}$ is a $D$-dimensional vector
Classification Algorithm

Notations (on binary classification)

Input space: $\mathcal{X} \subseteq \mathbb{R}^D$
Response space: $\mathcal{Y} := \{-1, 1\}$
Instance: $x \in \mathcal{X}$ is a $D$-dimensional vector
Hypothesis space: $\mathcal{H}$
Classification hypothesis: $f \in \mathcal{H}$, $f : \mathcal{X} \rightarrow \mathbb{R}$
Classification Algorithm

Notations (on binary classification)

Input space: $\mathcal{X} \subseteq \mathbb{R}^D$
Response space: $\mathcal{Y} := \{-1, 1\}$
Instance: $x \in \mathcal{X}$ is a $D$-dimensional vector
Hypothesis space: $\mathcal{H}$
Classification hypothesis: $f \in \mathcal{H}$, $f : \mathcal{X} \rightarrow \mathbb{R}$
Negative set (beginin): $\mathcal{X}^- := \{x \in \mathcal{X} | \text{sign}(f(x)) = -1\}$
Positive set (malicious): $\mathcal{X}^+ := \{x \in \mathcal{X} | \text{sign}(f(x)) = +1\}$
Classification Algorithm

Notations (on binary classification)

Input space: $\mathcal{X} \subseteq \mathbb{R}^D$
Response space: $\mathcal{Y} := \{-1, 1\}$
Instance: $x \in \mathcal{X}$ is a $D$-dimensional vector
Hypothesis space: $\mathcal{H}$
Classification hypothesis: $f \in \mathcal{H}$, $f : \mathcal{X} \to \mathbb{R}$
Negative set (beganin): $\mathcal{X}^- := \{x \in \mathcal{X} | \text{sign} (f(x)) = -1\}$
Positive set (malicious): $\mathcal{X}^+ := \{x \in \mathcal{X} | \text{sign} (f(x)) = +1\}$
Loss function: $V : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{0+}$
Goal

Given a training set \( S := \{(x_i, y_i) \mid x_i \in \mathcal{X}, y_i \in \mathcal{Y}\}_{i=1}^n \). Find the classifier \( f_S \in \mathcal{H} \) that performs best on some test set \( T \).

Solving Tikhonov regularization problem

\[
    f_S := \arg \min_{f} \gamma \sum_{i=1}^{n} V(y_i, f(x_i)) + \|f\|_{\mathcal{H}}^{2},
\]

where \( \gamma \in \mathbb{R}_{0+} \) is a fixed parameter for quantifying the trade off.
Exploratory Attack

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack

We assume the adversary ...

1. knows the dimension of input space;
2. attacks a fixed $f$;

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack

We assume the adversary ...

1. knows the dimension of input space;
2. attacks a fixed $f$;
3. only knows the family of $f$, e.g. $f$ is a convex-inducing classifier;
4. can observe $f(x)$ for any $x \in \mathcal{X}$ by a membership query;

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack

We assume the adversary ...

1. knows the dimension of input space;
2. attacks a fixed $f$;
3. only knows the family of $f$, e.g. $f$ is a convex-inducing classifier;
4. can observe $f(x)$ for any $x \in \mathcal{X}$ by a membership query;
5. knows $x^A \in \mathcal{X}_f^+$ and $x^- \in \mathcal{X}_f^-$;

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack

We assume the adversary ...

1. knows the dimension of input space;
2. attacks a fixed $f$;
3. only knows the family of $f$, e.g. $f$ is a convex-inducing classifier;
4. can observe $f(x)$ for any $x \in \mathcal{X}$ by a membership query;
5. knows $x^A \in \mathcal{X}_{f}^+$ and $x^- \in \mathcal{X}_{f}^-$;
6. designs the adversarial cost function;

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack

We assume the adversary ...

1. knows the dimension of input space;
2. attacks a fixed \( f \);
3. only knows the family of \( f \), e.g. \( f \) is a convex-inducing classifier;
4. can observe \( f(x) \) for any \( x \in X \) by a membership query;
5. knows \( x^A \in X^+_f \) and \( x^- \in X^-_f \);
6. designs the adversarial cost function;
7. has a *limited number* of probing opportunities.

- A trained & fixed classifier
- Find a disguised instance by querying
- Cost of “disguising”
- Use less queries
Exploratory Attack as $\ell_p$-norm Minimization

Original malicious instance: $\mathbf{x}^A \in \mathcal{X}^+$
Adversarial cost function: $g : \mathcal{X} \rightarrow \mathbb{R}_{0+}$
Focus on $g(\mathbf{x}) := \|\mathbf{x} - \mathbf{x}^A\|_{\ell_p}$
Exploratory Attack as $\ell_p$-norm Minimization

Original malicious instance: $x^A \in \mathcal{X}^+$
Adversarial cost function: $g : \mathcal{X} \rightarrow \mathbb{R}_{0+}$
Focus on $g(x) := \|x - x^A\|_{\ell_p}$

---

Exploratory Attack

Given $x^A$, $p$ and a membership oracle $f$, solve

$$\min_x \|x - x^A\|_{\ell_p} \quad \text{subject to} \quad x \in \mathcal{X}^-,\,$$

where $\mathcal{X}^-$ is specified by the membership oracle $f$. 
Exploratory Attack as $\ell_p$-norm Minimization

Original malicious instance: $x^A \in \mathcal{X}^+$
Adversarial cost function: $g : \mathcal{X} \rightarrow \mathbb{R}_{0+}$
Focus on $g(x) := \|x - x^A\|_{\ell_p}$

Exploratory Attack

Given $x^A$, $p$ and a membership oracle $f$, solve

$$\min_x \|x - x^A\|_{\ell_p} \quad \text{subject to} \quad x \in \mathcal{X}^-, \quad \mathcal{X}^+ \text{ is non-convex, } \mathcal{X}^- \text{ is convex and } p \geq 1.$$

where $\mathcal{X}^-$ is specified by the membership oracle $f$.

Find an instance $x$ that approximates $x^*$ with absolute error $\epsilon > 0$, i.e., $g(x) - g(x^*) \leq \epsilon$ in polynomial time.
Our Method

\[ x - \mathcal{P}^{(k)} \] at the \( k \) step

\[ x - \mathcal{P}^{(k+1)} \oplus x^A \] at the \( k + 1 \) step
Random Walks

Numerical difficulties of standard Hit-and-Run [Smith, 1984]

1. starting point is close to the boundary
2. convex body is not in isotropic position

- Workarounds
  - 1. Explicitly calculate the optimal direction [Kaufman, 1988]
  - 2. Implicitly maintain convex body in near-isotropic position [Bertsimas, 2004]
Theoretical Results

**Theorem 1**

In the $k^{\text{th}}$ iteration, the expected volume of $\mathcal{P}^{(k)}$ is at most

$$\mathbb{E}[\text{vol}(\mathcal{P}^{(k)})] \leq \left( \frac{D}{D + N} \right)^k \text{vol}(\mathcal{X}^-).$$

**Corollary 1**

Given a probability level $\xi > 0$, set

$$N \geq 2.2 \ln \frac{1}{\xi}.$$ 

Then, in each iteration algorithm cuts off more volume than the central-cut method with probability at least $1 - \xi$. 
Theoretical Results

**Theorem 2**

Given an initial instance \( \mathbf{x}^{(0)} \in \mathcal{X}^- \), the expected absolute error in the \( k^{th} \) iteration is at most

\[
\mathbb{E}[g(\mathbf{x}^{(k)}) - g^*] \leq \left( \frac{1}{N + 1} \right)^{\frac{k}{D}} \mathbb{E}[g(\mathbf{x}^{(0)}) - g^*].
\]

The expected number of iterations to find an \( \epsilon \)-optimal solution is at most

\[
k = \left\lceil \frac{D}{\ln(N + 1)} \ln \frac{g(\mathbf{x}^{(0)}) - g^*}{\epsilon} \right\rceil.
\]

**Corollary 2**

Given \( \epsilon > 0 \), algorithm can compute an instance \( \mathbf{x} \) such that \( g(\mathbf{x}) - g^* \leq \epsilon g^* \) in polynomial time.
On Real-world Data
Detecting Exploratory Attacks

![Graph showing detection of exploratory attacks with malicious and benign data points.](image)

- Malicious
- Benign

The graph illustrates the detection of exploratory attacks with initial, disguised, and original data points over time along dimensions Dim 1 and Dim 2.
Recap: Different Learning Settings

Besides “supervised” learning, do you know other learning settings?
Recap: Different Learning Settings

Besides “supervised” learning, do you know other learning settings?
Label Flips Attack

Given a training set, the adversary contaminates the training data through flipping labels.

**Adversarial Label Flip Attack**

The adversary aims to find a combination of label flips under a given *budget* so that a classifier trained on such data will have maximal classification error on some test data.
A Bilevel Formulation

Training set: $S := \{(x_i, y_i) \mid x_i \in \mathcal{X}, y_i \in \mathcal{Y}\}_{i=1}^n$

Indicate variables: $z_i \in \{0, 1\}, i = 1, \ldots, n$

Tainted labels: $y'_i := y_i (1 - 2z_i)$ so that if $z_i = 1$ then $y'_i = -y_i$
(i.e. flipped), otherwise $y'_i = y_i$

Flipping cost: $c_i \in \mathbb{R}_{0+}$

Finding the optimal label flips

Given a test set $T$ and a budget $C$, solve

$$\max_{z} \sum_{(x, y) \in T} V(y, f_{S'}(x)),$$

s.t. $f_{S'} \in \arg \min_f \gamma \sum_{i=1}^n V(y'_i, f(x_i)) + \|f\|_H^2,$$

$$\sum_{i=1}^n c_i z_i \leq C, \quad z_i \in \{0, 1\}, \quad i = 1, \ldots, n.$$
A Relaxed Formulation

\[ \min_{q,f} \gamma \sum_{i=1}^{2n} q_i [V(y_i, f(x_i)) - V(y_i, f_S(x_i))] + \| f \|_\mathcal{H}^2, \quad (1) \]

s.t. \[ \sum_{i=n+1}^{2n} c_i q_i \leq C, \]
\[ q_i + q_{i+n} = 1, \quad i = 1, \ldots, n, \]
\[ q_i \in \{0, 1\}, \quad i = 1, \ldots, 2n. \]
Label Flip Attack on SVMs

Alternately solving the following two problems

\[
\min_{w, \epsilon, b} \gamma \sum_{i=1}^{2n} q_i \epsilon_i + \frac{1}{2} \|w\|^2
\]
\[
\text{s.t. } y_i(w^\top x_i + b) \geq 1 - \epsilon_i, \quad \epsilon_i \geq 0, \quad i = 1, \ldots, 2n.
\]  

\[
\min_{q} \gamma \sum_{i=1}^{2n} q_i (\epsilon_i - \xi_i)
\]
\[
\text{s.t. } \sum_{i=n+1}^{2n} c_i q_i \leq C,
\]
\[
q_i + q_{i+n} = 1, \quad i = 1, \ldots, n,
\]
\[
0 \leq q_i \leq 1, \quad i = 1, \ldots, 2n.
\]
**Synthetic Data**

(a) Synthetic data

- Linear pattern
  - Linear SVM: 1.8%
  - RBF-SVM: 3.2%
- Parabolic pattern
  - Linear SVM: 23.5%
  - RBF-SVM: 5.1%

(b) No Flips

- Linear SVM: 1.9%
- RBF-SVM: 4.0%

(c) Random

- Linear SVM: 6.9%
- RBF-SVM: 3.5%

(d) Nearest

- Linear SVM: 9.5%
- RBF-SVM: 26.5%

(e) Furthest

- Linear SVM: 21.8%
- RBF-SVM: 32.4%

(f) ALFA

- Linear SVM: 2.8%
- RBF-SVM: 3.4%
An application

Researchers in adversarial learning are good guys

Q: Sounds very counter-productive?
A: If we prove a learning algorithm is not secure, then use it with your own risk! *Don’t use it to control nuclear missiles!*

Q: Name an application?
A: Amazon’s Mechanical Turk – A crowd-sourcing platform
Amazon’s Mechanical Turk
a marketplace for work that requires human intelligence
Amazon’s Mechanical Turk

a marketplace for work that requires human intelligence

- **Requester** submit their tasks
- **Worker** select the task
- Sophisticate algorithm for assigning the task
- Detecting lazy-worker, adversary, haters use *wisdom of crowds*
- Only real contributors get paid
Amazon’s Mechanical Turk
a marketplace for work that requires human intelligence

- **Requester** submit their tasks
- **Worker** select the task
- Sophisticate algorithm for assigning the task
- Detecting lazy-worker, adversary, haters use *wisdom of crowds*
- Only real contributors get paid

Probably in the future...
- Requester *do not* need to provide the groundtruth
- The expertise level of each worker can be more accurately modeled
- Active sampling saves the cost of requester
Summary

• Security problem in machine learning
• What is adversarial learning?
• Why do we study it?
• What is the exploratory attack?
  • How to defense?
• What is the causative attack?
• A positive application – Amazon’s Mechanical Turk

Possible topics for Master thesis