Adversarial Machine Learning

CS229

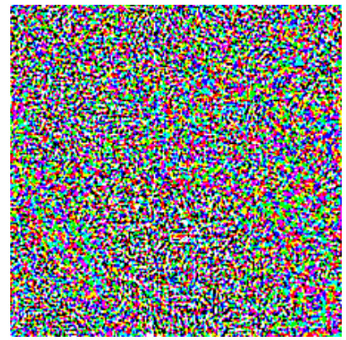
Tengyu Ma

Security Problems in Machine Learning

- Adversarial training data
  - ML training data are often crowd-sourced or crawled from the web
  - Can malicious training data destroy the model, or create backdoors?
- Adversarial test data
  - Adversarial test example can fool the classifier
- Data privacy
  - If a model learned partially from data on your cell phone is made public, can others extract your personal information from the model?
- Note: issues are not necessarily specific to modern ML; they existed before as well, but attracted less attention because ML didn't work as well as it does today.

Adversarial Examples at Test Time

$+ .007 \times$



$=$

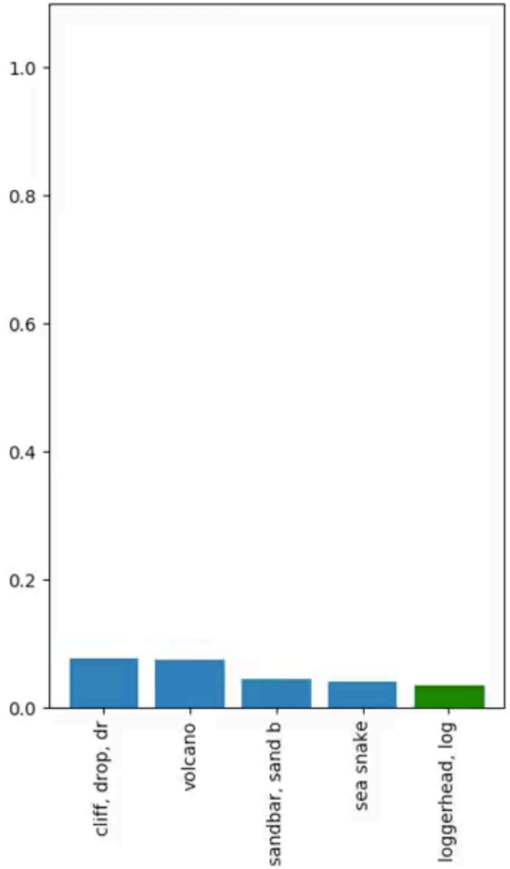
"panda"
57.7% confidence

"gibbon"
99.3 % confidence

Image credit:
Above: Explaining And Harnessing Adversarial Examples. Ian J. Goodfellow, Jonathon Shlens & Christian Szegedy, 2015
Right: Wikipedia

3D Adversarial Examples

- A turtle that is almost always classified as a rifle



[Synthesizing Robust Adversarial Examples Anish Athalye, Logan Engstrom, Andrew Ilyas, Kevin Kwok, 2018]
Video link: https://www.youtube.com/watch?v=YXy6oX1iNoA&feature=youtu.be

Formulation

- Supervised learning with binary classification
  - $\mathcal{X} = \mathbb{R}^d, \mathcal{Y} = \{-1,1\}$
  - $f: \mathcal{X} \to \mathbb{R}$
- Training distributions $\mathcal{D}$, clean test distribution $\mathcal{D}$
- Clean test accuracy: $\Pr_{(x,y)\sim\mathcal{D}}[1(yf(x) > 0)]$
- Attack/threat model: can perturb x to get adversarial example $\hat{x}$
  - Commonly-studied attack model: $\hat{x} = x + \Delta$ where $||\Delta||_\infty \leq \delta$
  - For small δ (say $\delta = 0.1$ when coordinates of x has average scale 1), such perturbation often does not affect human classification
- Attacker's goal: find $\hat{x}$ such that $yf(\hat{x}) < 0$
- Defender's goal: maximize the robust test accuracy
  - $\Pr_{(x,y)\sim\mathcal{D}}[\forall \hat{x} = x + \Delta \text{ with } ||\Delta||_\infty \leq \delta, \text{s.t.}, 1(yf(\hat{x}) > 0)]$

Attack Algorithms

Fast gradient sign method (FGSM)

- Let $\ell\big((x, y); \theta\big)$ be the loss function for training example (x, y)
- Recall that small loss $\Rightarrow f(x)$ is correct
- Attack: $\hat{x} = x + \delta \cdot \mathrm{sign}(\nabla_x \ell\big((x, y); \theta\big))$

Projected gradient descent (PGD)

- Solve the optimization problem below by projected gradient ascent

$$\max \ell((\hat{x}, y); \theta)$$

$$\text{s.t. } ||\hat{x} - x||_\infty \le \delta$$

Defense: Adversarial Training

Idea: solving the min-max problem

$$\underbrace{argmin_{\theta}\frac{1}{n}}_{\text{The parameters for which no adversarial examples increases the loss much}}\sum_{i=1}^{n}\underbrace{\max_{||\Delta^{(i)}||_{\infty}\leq\delta}\ell\left(\left(x^{(i)}+\Delta^{(i)},y^{(i)}\right);\theta\right)}_{\text{the “best” adversarial examples the attacker can find}}$$

Computational challenge:

- the max cannot be evaluated exactly
- heuristic: iteratively update $\Delta^{(i)'}s$ and θ

Am empirically strong defense; but hard to scale to large datasets due to computational overheads