Adversarial Examples

The (never-ending) road to robustness in deep learning.

Paul KRZAKALA

Group Meeting presentation

Our story begins...

Once upon a time in 2013...

- Its a great time to be Yann Lecun!
- Neural Nets are getting deeper!
- Neural Nets are getting better!

Everything is going great for Deep Learning! Until...

$$\min\{||\eta|| \ / \ g(x+\eta) \neq g(x)\} \quad ? \tag{1}$$

Adversarial Example

Intriguing properties of neural networks

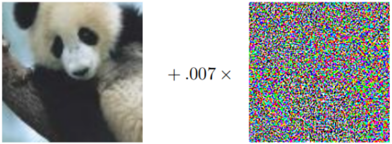
Christian Szegedy Google Inc.
Wojciech Zaremba New York University
Ilya Sutskever Google Inc.
Joan Bruna New York University
Dumitru Erhan Google Inc.
Ian Goodfellow University of Montreal
Rob Fergus New York University, Facebook Inc.

"panda"
57.7% confidence

Adversarial Example

Intriguing properties of neural networks

Christian Szegedy Google Inc.
Wojciech Zaremba New York University
Ilya Sutskever Google Inc.
Joan Bruna New York University
Dumitru Erhan Google Inc.
Ian Goodfellow University of Montreal
Rob Fergus New York University, Facebook Inc.



"panda"
57.7% confidence

"nematode"
8.2% confidence

Adversarial Example

Intriguing properties of neural networks

Christian Szegedy
Google Inc.

Wojciech Zaremba
New York University

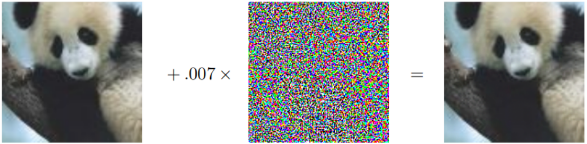
Ilya Sutskever
Google Inc.

Joan Bruna
New York University

Dumitru Erhan
Google Inc.

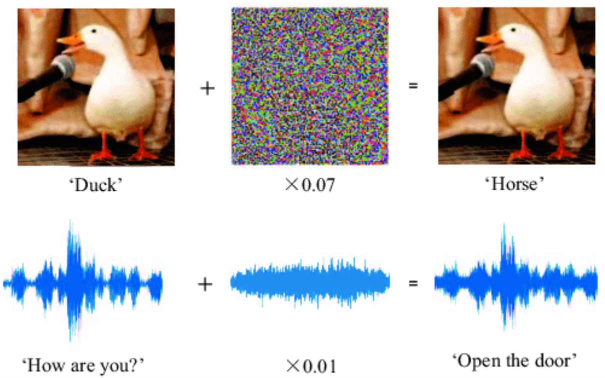
Ian Goodfellow
University of Montreal

Rob Fergus
New York University
Facebook Inc.



"panda" 57.7% confidence | "nematode" 8.2% confidence | "gibbon" 99.3 % confidence

Adversarial Examples



Strange properties

The very existence of adversarial examples is strange but they also exhibit strange properties:

- Omnipresence (across architecture, datatype, instances)
- High Confidence error
- Transferability (black box attack)

In this presentation

I. Definitions

II. Attacks & Defences

III. Origins of adversarial examples

I. Definitions

Notations

Setting = multiclass classification: input space $\mathcal{X}$, K classes

We consider deep neural nets

$$f : \mathcal{X} \to \Sigma_K$$

and the associated classifier $g : \mathcal{X} \to [1, K]$

$$g(x) = \arg\max_{i \in K} [f(x)]_i$$

Definition of an adversarial example

Assuming k is the true class of x and $g(x) = k$

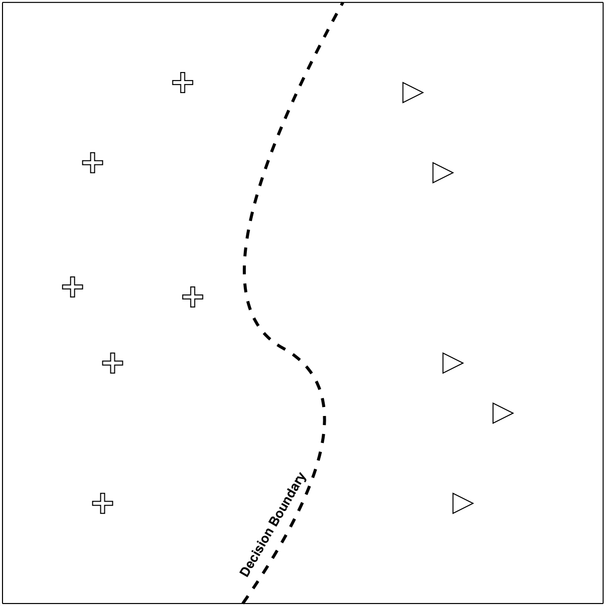
Robustness radius:

$$\epsilon = \min\{||\eta|| \ / \ g(x+\eta) \neq k\} \tag{2}$$

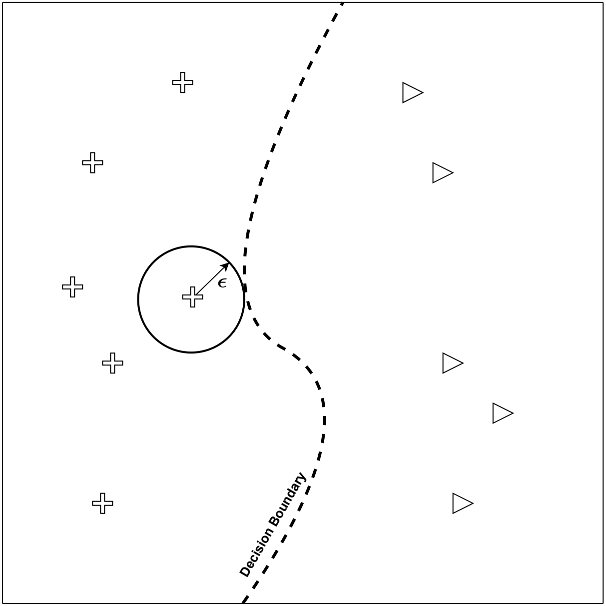
Bounded adversarial attack:

$$x' = \underset{||x'-x|| \leq \epsilon}{\arg\min} \, [f(x)]_k \tag{3}$$

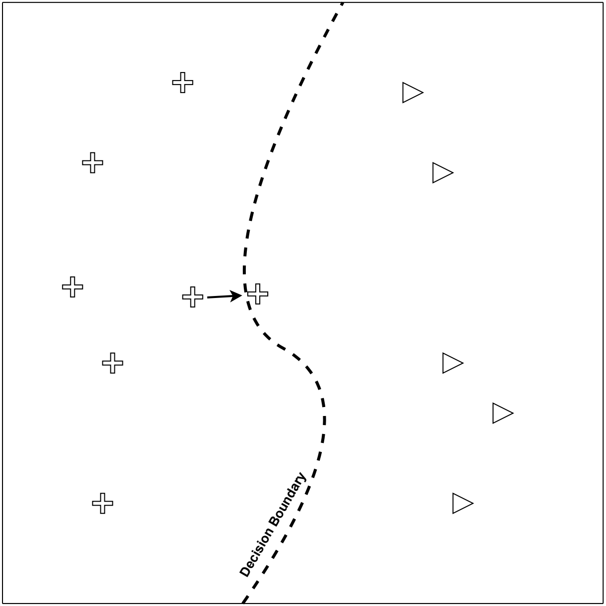
2D representation



2D representation



2D representation



Adversarial Robustness

Classification error:

$$\mathcal{R}_{std} = \mathbb{E}(\mathbb{1}[g(x) \neq k]) \tag{4}$$

Adversarial error:

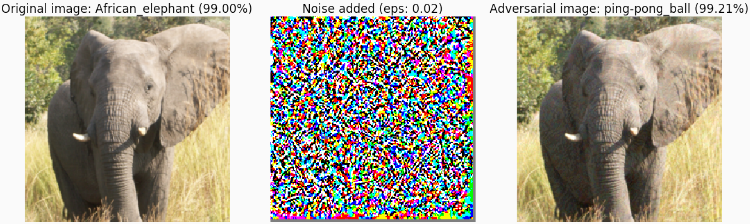
$$\mathcal{R}_{rob} = \mathbb{E}(\max_{||\eta|| \leq \epsilon} \mathbb{1}[g(x+\eta) \neq k]) \tag{5}$$

The scale of the problem

Consider images in $[0,1]^{3xNxN}$, ex: ImageNet.

For a typical choice $||\eta||_\infty \leq \epsilon = \frac{4}{255}$ or $||\eta||_2 \leq \epsilon = 0.5$

$$\mathcal{R}_{std} \ll \mathcal{R}_{rob} \tag{6}$$



II. Attacks & Defences

Definitions

Attacks = solvers for

$$A(g, x, k, ||\cdot||, \epsilon) \approx \underset{||x-x'||\leq\epsilon}{\arg\min}\,[f(x')]_k \tag{7}$$

$\rightarrow$ for clarity we simply denote $A(x)$.

Defence = method (architecture, learning algorithm...) to minimize

$$\min_{g\in\mathcal{H}} \mathcal{R}_{rob}(g) = \min_{g\in\mathcal{H}} \mathbb{E}(\max_{||\eta||\leq\epsilon} \mathbb{1}[g(x+\eta)\neq k]) \tag{8}$$

$\rightarrow$ this is a saddle point problem.

Here comes trouble...

$\mathcal{R}_{rob}$ can only be estimated **given** an attack

$$\mathcal{R}_{rob} \approx \mathbb{E}(\mathbb{1}[g(A(x) \neq k]) = \mathcal{R}_{rob}^{A} \tag{9}$$

Actually $\mathcal{R}_{rob} \leq \mathcal{R}_{rob}^{A}$

This can give a 'false sense of security'

Architecture A is all your need for adversarial robustness.

Abstract

In this paper, we introduce a brilliant new architecture that totally solves the probleme of adversarial examples. We achieve this result by turbo-rotating the ReLU activations in the Fourier space (as defined by the appropriate kernel).

The Attack/Defence arm race

Introducing Attack B, a new adversarial attack that bypasses the defences of architecture A.

Abstract

We introduce a new adversarial examples generation technique that can fool even architecture A which was belied to be robust to adversarial attack. Our method is based on mirror double projected gradient descent on the dual of the network.

This time I swear we found a way to train adversarially robust networks.

Abstract

In this paper, we introduce a new learning process that yields adversarially robust deep networks. We achieve an unprecedented robust accuracy by introducing images of my vacations in the alps to the training set, pretty sure it works.

Actually, you did not. Introducing 5 new adversarial attacks that bypasses your defence.

Abstract

Steve et al. introduced a learning process yielding network robust to attack B. In this paper we introduce a new set of attack, all bypassing this defence mechanism. It was quite easy actually, too bad Steve.

An example of attack

$\rightarrow$ Consider binary classification:

$$g(x) = 1 \iff f(x) > 0$$

If the true class is 1, the adversarial attack amount to compute

$$\min_{||\eta|| \leq \epsilon} f(x + \eta)$$

Using the linear approximation $f(x + \eta) \approx f(x) + \langle \nabla_x f(x), \eta \rangle$

Thus

$$\eta^* \approx \min_{||\eta|| \leq \epsilon} \langle \nabla_x f(x), \eta \rangle$$

For L_2: $\eta^* = -\epsilon \frac{\nabla_x f(x)}{||\nabla_x f(x)||}$

For L_∞: $\eta^* = -\epsilon \mathtt{sign}(\nabla_x f(x))$

Stronger attacks

More generally if the loss the networks tries to minimize is

$$\ell(f(x), y)$$

An attack can be computed by maximizing

$$\max_{||\eta|| \leq \epsilon} \ell(f(x), y)$$

Typically using n steps of projected gradient descent (PGD-n).

An example of defence

Recall that the goal of a defence is to minimize:

$$\mathcal{R}_{rob}(g) = \mathbb{E}(\max_{||\eta|| \leq \epsilon} \mathbb{1}[g(x+\eta) \neq k]) \tag{10}$$

We can apply the classical convex + empirical relaxations + denote θ the parameters of the model

$$\mathcal{L}_{rob}(\theta, x_1, ...x_N) = \frac{1}{N} \sum_{i=1}^{N} \max_{||x'_i - x_i|| \leq \epsilon} \ell(f_\theta(x'_i), y_i) \tag{11}$$

In comparison the standard loss is

$$\mathcal{L}_{std}(\theta, x_1, ...x_N) = \frac{1}{N} \sum_{i=1}^{N} \ell(f_\theta(x_i), y_i) \tag{12}$$

Adversarial training

Under mild conditions

$$\nabla_\theta \mathcal{L}_{rob}(\theta, x_1, ...x_N) = \nabla_\theta \mathcal{L}_{std}(\theta, x'_1, ...x'_N) \tag{13}$$

where $x'_i = \underset{||x'_i - x_i|| \leq \epsilon}{\arg\max} \; \ell(f_\theta(x'_i), y_i)$

$\rightarrow$ Standard Training + feed the network with adversarial attacks

$\rightarrow$ This can be seen as a form of ”active” data augmentation

Adversarial training

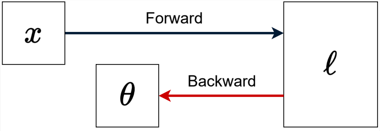


Figure 1: Standard Training

.

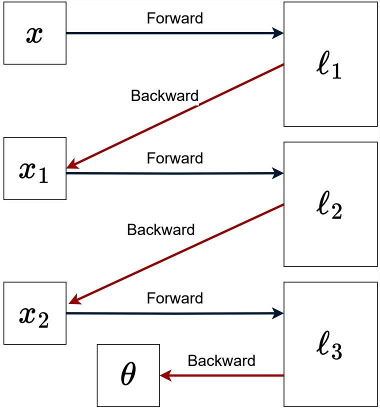


Figure 2: Adversarial Training

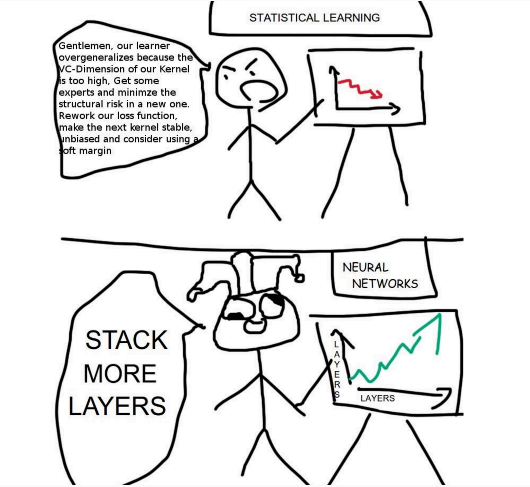
Limitations

Limitations of adversarial training

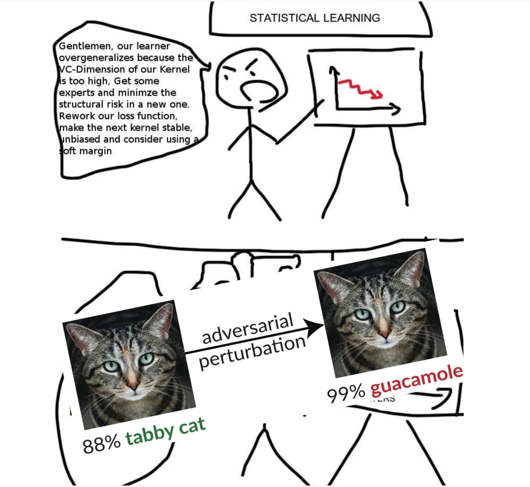
- Typically x10 to x100 more expensive
- Weak adversarial attack at test time $\rightarrow$ vulnerability to strong attack at test time
- Trade-off std vs robust accuracy

III. Origins of adversarial examples

Explanation 1: The divine punishment



Explanation 1: The divine punishment



Explanation 2: Linearity

Let $f_\theta : \mathbb{R}^d \to \mathbb{R}$ be a linear model, i.e. $f_\theta(x) = \langle \theta, x \rangle$.

Then

$$\max_{||\eta||_p \leq \epsilon} |f(x + \eta) - f(x)| = \epsilon ||\theta||_q \tag{14}$$

where q is the dual of p i.e. $\frac{1}{p} + \frac{1}{q} = 1$.

For instance

- $p = \infty \implies q = 1$
- $p = 2 \implies q = 2$

Deep learning $\implies$ d very large (image net: $256 \times 256 \times 3 = 196608$)

$\implies$ $||\theta||_q$ very large !

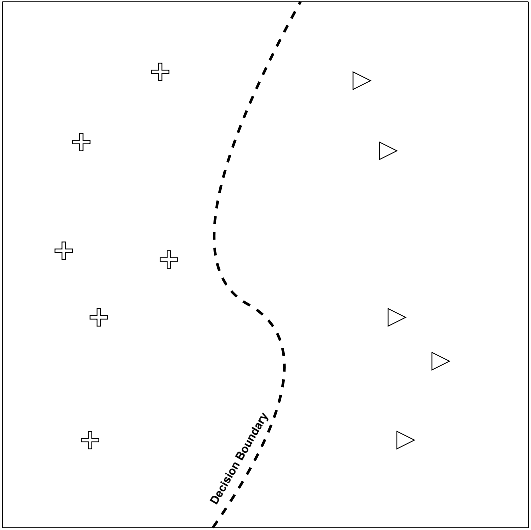
Explanation 2: Linearity

Notes:

- Link with explanation 1: no dimentionnality reduction in deep learning
- In high dimension $||x||_q$ and $||x||_{q'}$ can be very different hence the vulnerability to specific perturbations

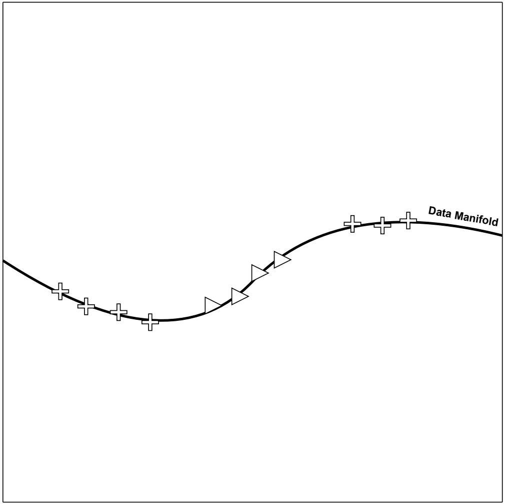
Explanation 3: data manifold

Standard representation of the data d dimensional data.
For $d = 2$:



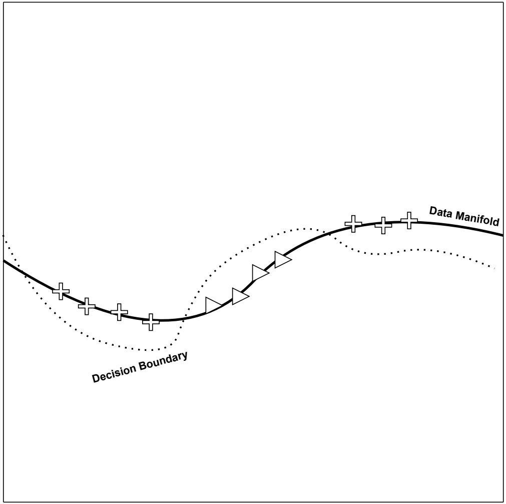
Explanation 3: data manifold

High dimensional data tend to lie on a m dimensional manifold. Typically $m \ll d$. For $m = 1, d = 2$:



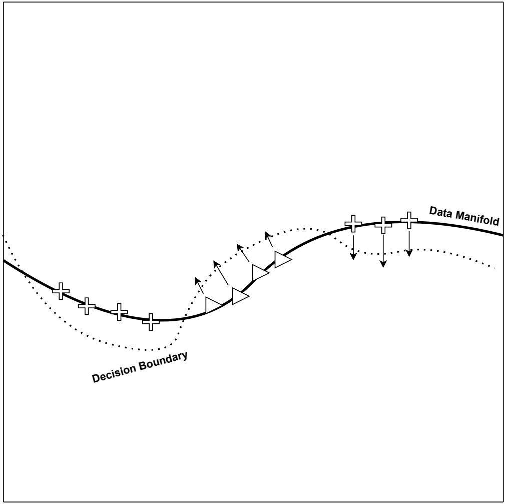
Explanation 3: data manifold

Hypothesis: the decision boundary is too close to the data manifold (the network is lazy).

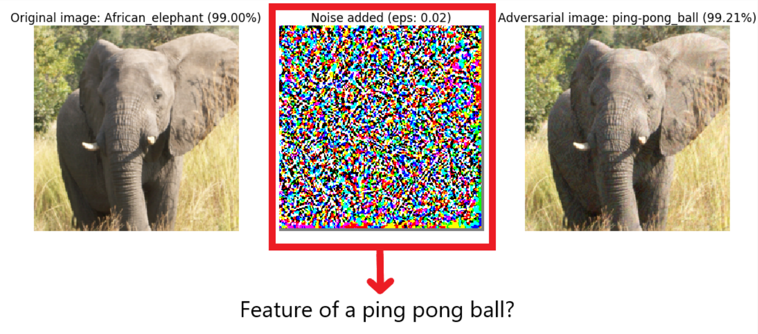


Explanation 3: data manifold

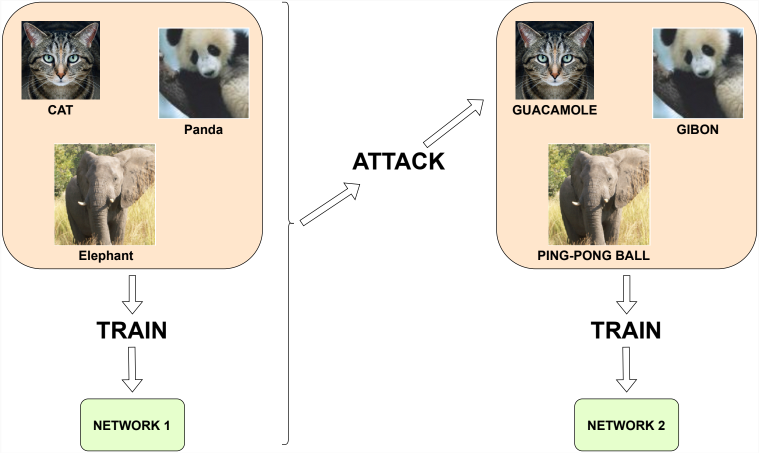
Hypothesis: Adversarial attacks are orthogonal to the data manifold



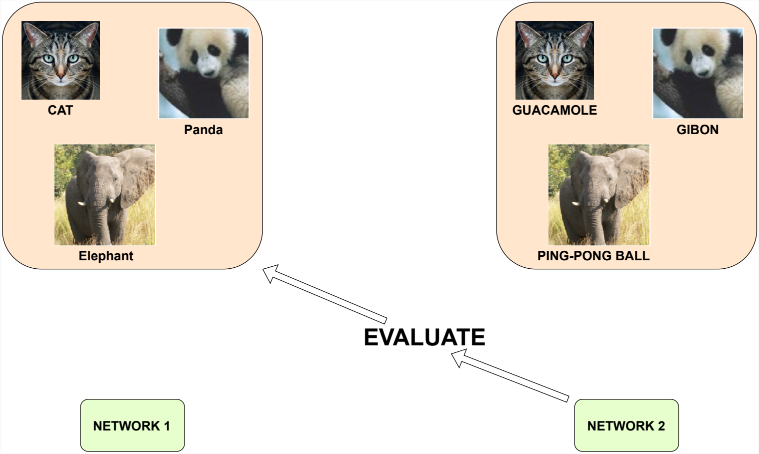
"Adversarial Examples are not bugs, they are features"



Testing explanation 4:



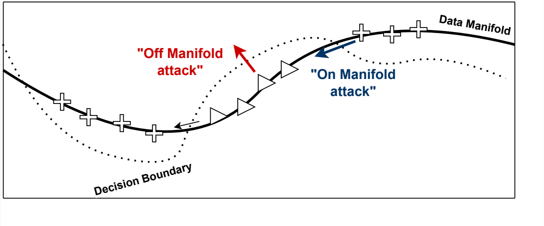
Testing explanation 4:



Non-robust features

Conclusions:

- There exist non-robust (in the human sense) but statistically useful features
- This may explain transferability of adversarial examples
- This may explain why the trade-off between robustness and accuracy
- This may not explain all adversarial examples



Takeaway on the origins of adversarial examples

- Adversarial examples arise from high dimension of the data (more than from the network itself)
- The definition of a ”small perturbation” is ill-posed, there is a misalignment between ”small for a human” and ”small for a model”
- There are different phenomenon at play