Vulnerabilities of Deep Neural Networks

Alexey Gruzdev
a.s.gruzdev@yandex.ru

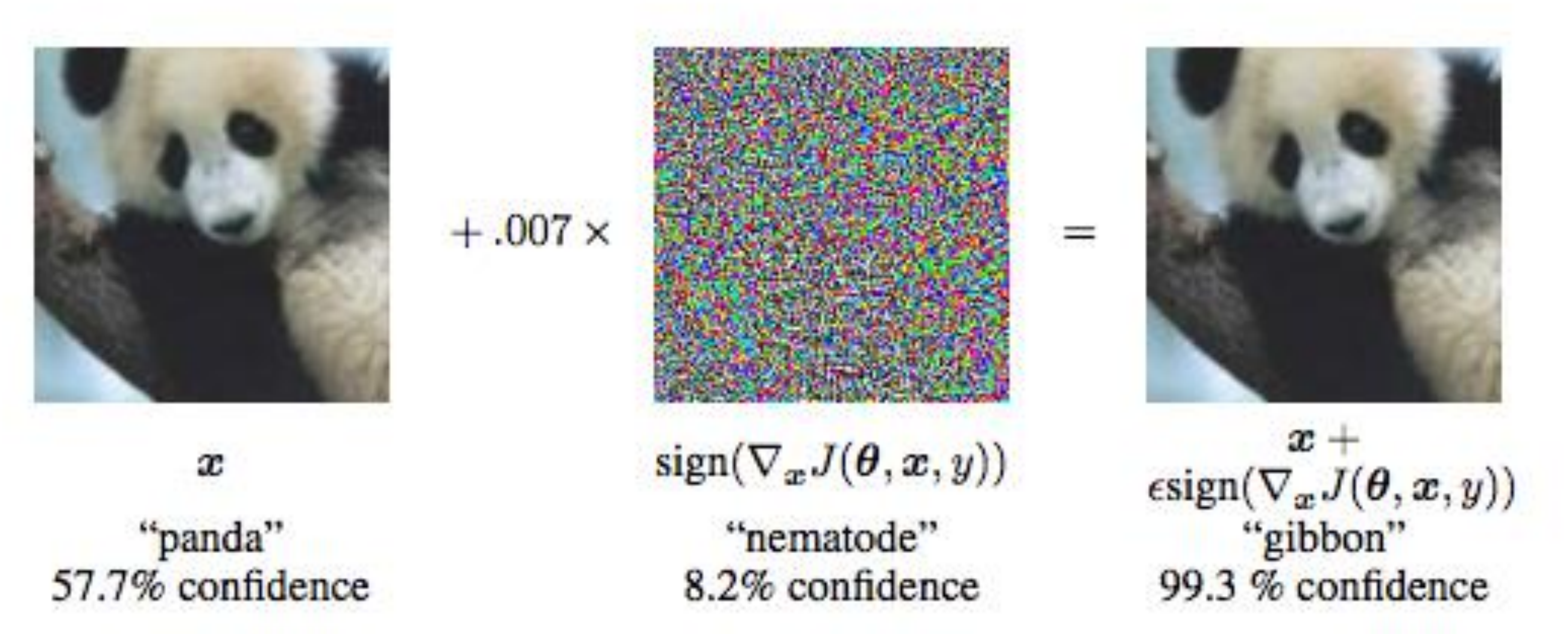
HSE, Summer, 2017

Can we fool the Neural Networks?

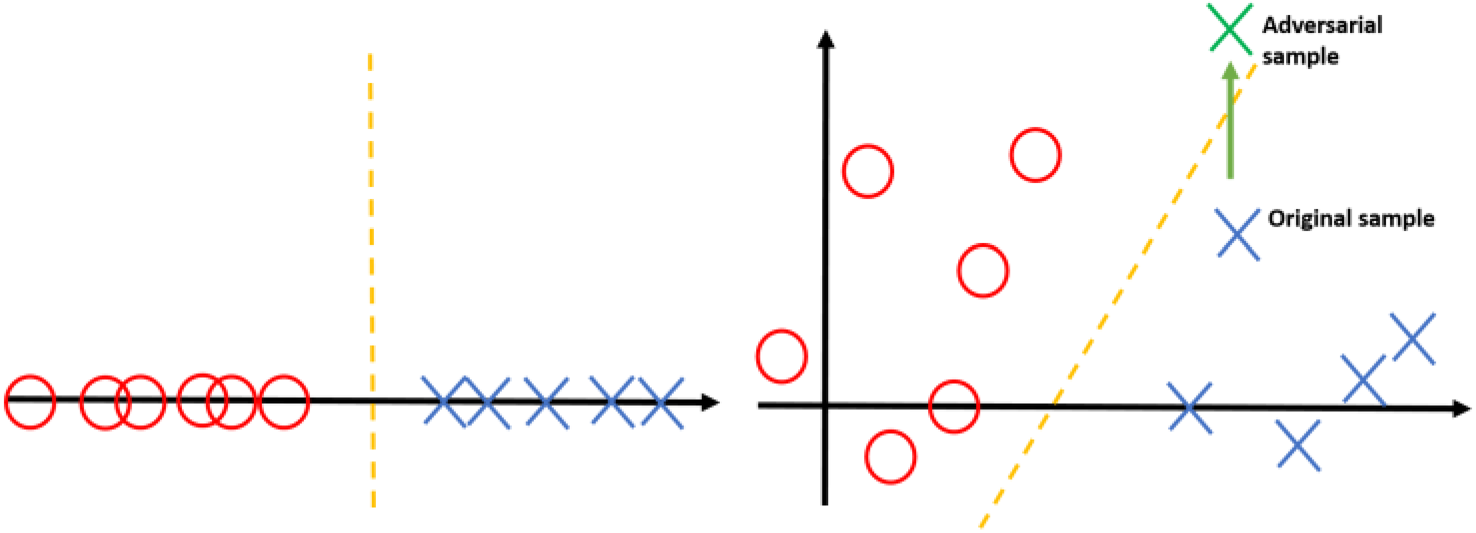
Short Answer: Yes, we can

Long Answer: This work

Intriguing properties of neural networks - Feb 2014



Adversarial intuition



Ok. But How?

- **Fast sign (gradient sign) method:**

$$\boldsymbol{X}^{adv} = \boldsymbol{X} + \epsilon\,\mathrm{sign}\big(\nabla_X J(\boldsymbol{X}, y_{true})\big)$$

- **Basic iterative method:**

$$\boldsymbol{X}_0^{adv} = \boldsymbol{X}, \quad \boldsymbol{X}_{N+1}^{adv} = Clip_{X,\epsilon}\Big\{\boldsymbol{X}_N^{adv} + \alpha\,\mathrm{sign}\big(\nabla_X J(\boldsymbol{X}_N^{adv}, y_{true})\big)\Big\}$$

- **Iterative least-likely method:**

$$\boldsymbol{X}_0^{adv} = \boldsymbol{X}, \quad \boldsymbol{X}_{N+1}^{adv} = Clip_{X,\epsilon}\left\{\boldsymbol{X}_N^{adv} - \alpha\,\mathrm{sign}\left(\nabla_X J(\boldsymbol{X}_N^{adv}, y_{LL})\right)\right\}$$

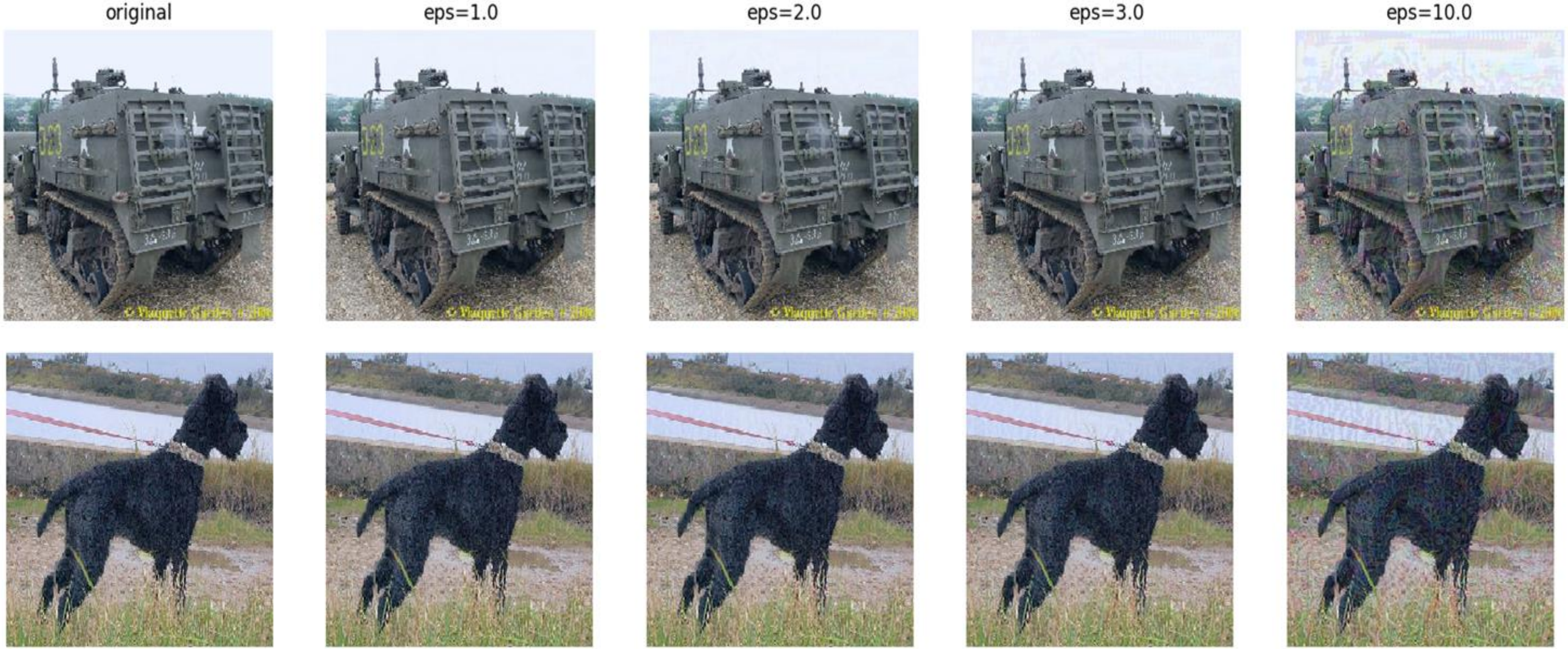
Clipping procedure

$Clip_{X,\epsilon}\left\{\boldsymbol{X}'\right\}$ - function which performs per-pixel clipping of the image $\boldsymbol{X}'$, so the result will be in L_∞ ϵ-neighbourhood of the source image $\boldsymbol{X}$. The exact clipping equation is as follows:

$$Clip_{X,\epsilon}\left\{\boldsymbol{X}'\right\}(x,y,z) = \min\Big\{255, \boldsymbol{X}(x,y,z)+\epsilon, \max\big\{0, \boldsymbol{X}(x,y,z)-\epsilon, \boldsymbol{X}'(x,y,z)\big\}\Big\}$$

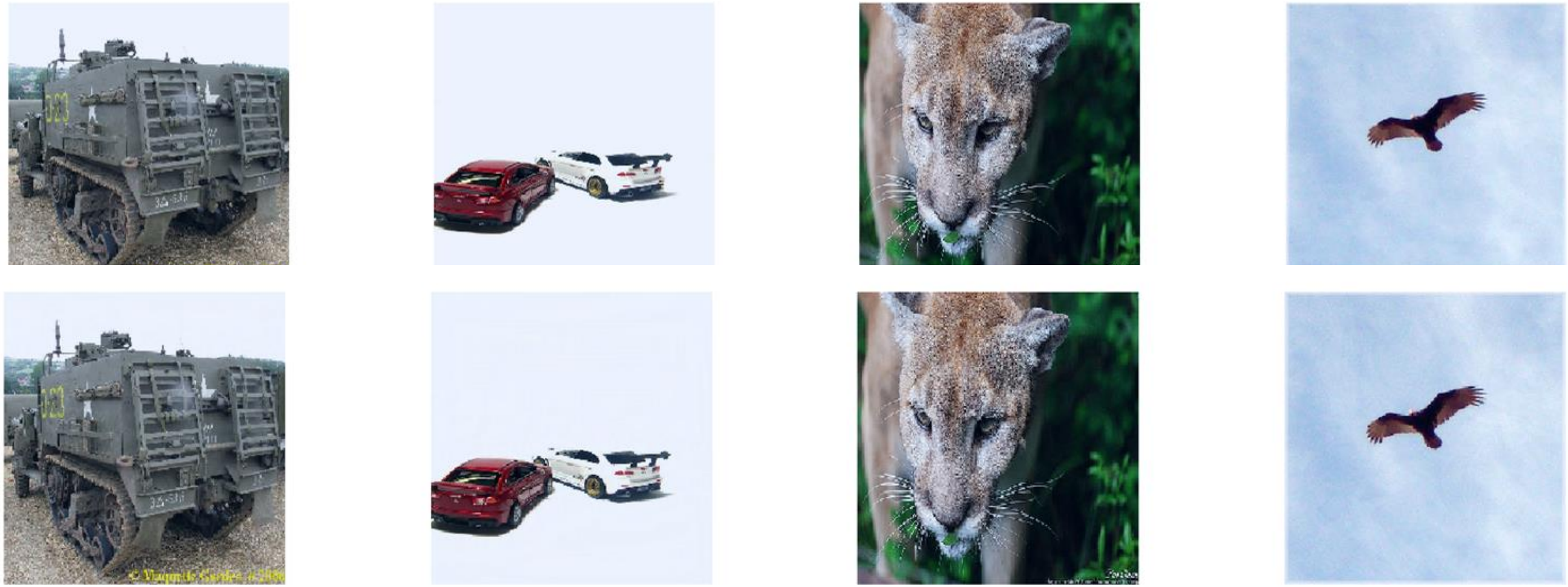
where $\boldsymbol{X}(x,y,z)$ is the value of channel z of the image $\boldsymbol{X}$ at coordinates (x,y).

Let’s observe FGSM practically



Let’s observe FGSM practically

Adversarial examples vs Original (FGSM)



Wait, we know all weights in DNN, what if we don’t?

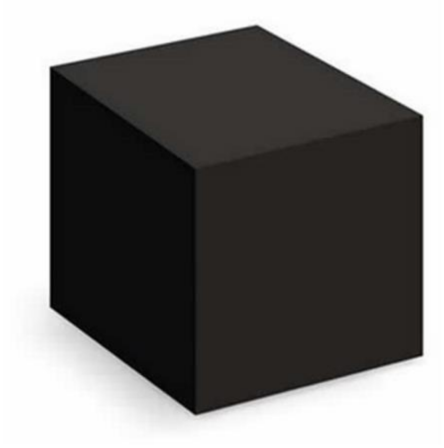
Q: What if we don’t have access to model weights/ architecture & training data?

A: Let’s approximate this model by another neural network and construct adversarial examples for `substitution`!

Black-Box attack

Black-Box attack:

1. Train substitute model (maybe not a neural network) on data with labels from oracle model
2. Construct adversarial examples for substitute model
3. Apply these examples for targeted model.



Results for ImageNet dataset

Attack Method	VGG-16	AlexNet	ResNet-50	Inception v3
FGSM	0.85/0.56/0.61	0.56/0.22/0.33	0.88/0.53/0.62	**0.92**/0.01/0.07
Black-Box	0.85/0.61	0.56/0.43	0.88/0.69	**0.92**/0.19

- FGSM is better for generating adversarial examples than Black-Box
- Inception v3 - best performer is much more vulnerable

- No architectures were stable

Results for CIFAR-10, CIFAR-100

Attack Method	VGG-16	ResNet-50
FGSM	0.79/0.03/0.36	**0.81**/0.05/0.43
Black-Box	0.79/0.44	**0.81**/0.51

Attack Method	VGG-16	ResNet-50
FGSM	0.64/0.15/0.36	**0.67**/0.29/0.54
Black-Box	0.64/0.41	**0.67**/0.38

<u>Motivation:</u>

- For ImageNet task input resolution is 224x224, what if you have less pixels to manipulate
- We still were able to drop the accuracy!

Transferability check

Attack Method	AlexNet	ResNet-50	Inception v3
FGSM	0.56/0.10/0.23	0.88/0.57/0.58	**0.92**/0.03/0.06

Attack Method	VGG-16	ResNet-50	Inception v3
FGSM	0.85/0.4/0.49	0.88/0.45/0.46	**0.92**/0.04/0.07

- Adversarial examples preserve the transferability - images generated to hack 1 model can hack others as well
- What’s wrong with InceptionV3?!

Vulnerabilities of Deep Neural Networks

Alexey Gruzdev
a.s.gruzdev@yandex.ru

HSE, Summer, 2017