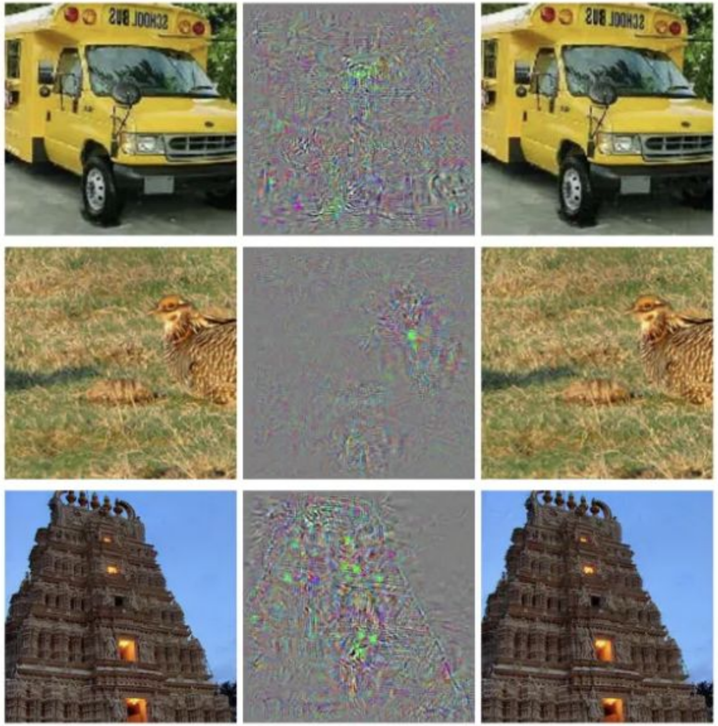
Advanced Machine Learning

Likhit Nayak

Adversarial Examples



Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

Adversarial Examples

Assumption in neural network training:

For a small enough radius ***r*** in the vicinity of a given input ***x***, an input ***x* + *r*** satisfying $\|\boldsymbol{r}\| < \boldsymbol{\epsilon}$ ($\epsilon > 0$) will predict the correct class

Disproved:

The assumption does not hold true, and it is proved by generating adversarial samples with imperceptibly small perturbations that make the model mispredict classes.

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

Adversarial Examples

Consider a neural network classifier ***f*** that maps an input image $\boldsymbol{x} \in \mathbb{R}^m$ to a set of labels $\{1 \ldots k\}$ — $f : \mathbb{R}^m \rightarrow \{1 \ldots k\}$

$$\text{Minimize } \|r\|_2 \text{ subject to:}$$

1. $f(x + r) = l$
2. $x + r \in [0, 1]^m$

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

Cross-model generalization

	FC10(10^{-4})	FC10(10^{-2})	FC10(1)	FC100-100-10	FC200-200-10	AE400-10	Av. distortion
FC10(10^{-4})	100%	11.7%	22.7%	2%	3.9%	2.7%	0.062
FC10(10^{-2})	87.1%	100%	35.2%	35.9%	27.3%	9.8%	0.1
FC10(1)	71.9%	76.2%	100%	48.1%	47%	34.4%	0.14
FC100-100-10	28.9%	13.7%	21.1%	100%	6.6%	2%	0.058
FC200-200-10	38.2%	14%	23.8%	20.3%	100%	2.7%	0.065
AE400-10	23.4%	16%	24.8%	9.4%	6.6%	100%	0.086
Gaussian noise, stddev=0.1	5.0%	10.1%	18.3%	0%	0%	0.8%	0.1
Gaussian noise, stddev=0.3	15.6%	11.3%	22.7%	5%	4.3%	3.1%	0.3

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

Cross-training generalization

	FC100-100-10	FC123-456-10	FC100-100-10’
Distorted for FC100-100-10 (av. stddev=0.062)	100%	26.2%	5.9%
Distorted for FC123-456-10 (av. stddev=0.059)	6.25%	100%	5.1%
Distorted for FC100-100-10’ (av. stddev=0.058)	8.2%	8.2%	100%
Gaussian noise with stddev=0.06	2.2%	2.6%	2.4%

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

How unstable are neural networks?

If minor perturbations to the inputs change the model's output to the point that it becomes incorrect, then the model is not stable.

Lipschitz continuity:

Any function where the slope of the line joining any two points on this function is not greater than a real number is called a **Lipschitz continuous function**. This real number is called the **Lipschitz constant**.

$$|f(x_1) - f(x_2)| \leq K|x_1 - x_2|$$

How unstable are neural networks?

The unstability of a neural network can be understood by looking at the Lipschitz constant of each layer ***k = 1 . . . K,*** as given by:

$$\forall x, r \quad \|\varphi_k(x; W_k) - \varphi_k(x + r; W_k)\| \leq L_k \|r\|$$

where $\boldsymbol{\varphi_k}$ is the output of the $\boldsymbol{k^{th}}$ **layer**, and ***L***□ is the Lipschitz constant for the $\boldsymbol{k^{th}}$ **layer.** So for the entire network of ***K*** layers,

$$\|\varphi(x) - \varphi(x + r)\| \leq L\|r\| \quad where\ L = \prod_{k=1}^{K} L_k$$

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

How unstable are neural networks?

Layer	Size	Stride	Upper bound
Conv. 1	$3 \times 11 \times 11 \times 96$	4	2.75
Conv. 2	$96 \times 5 \times 5 \times 256$	1	10
Conv. 3	$256 \times 3 \times 3 \times 384$	1	7
Conv. 4	$384 \times 3 \times 3 \times 384$	1	7.5
Conv. 5	$384 \times 3 \times 3 \times 256$	1	11
FC. 1	9216×4096	N/A	3.12
FC. 2	4096×4096	N/A	4
FC. 3	4096×1000	N/A	4

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

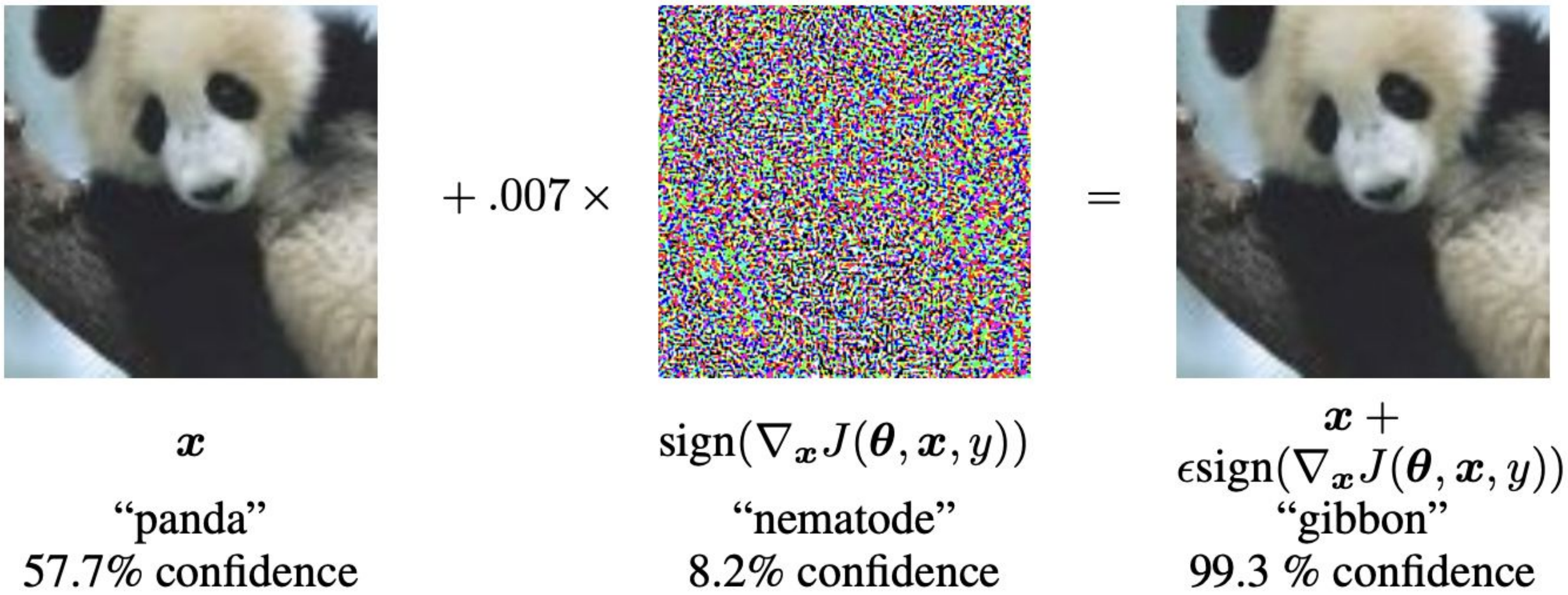
How unstable are neural networks?

Properties of Lipschitz constant for a neural network:

- Small bounds guarantee that no such adversarial examples can appear.
  - This suggests a simple regularization of the parameters, consisting in penalizing each upper Lipschitz bound, which might help improve the generalisation error of the networks
- Large bounds do not automatically translate into existence of adversarial examples
- They don’t explain why adversarial examples generalize across different hyperparameters or training sets.

Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

Adversarial Training



Adversarial Training - Loss function

- Adversarial examples are a result of models being too linear, rather than too nonlinear.
- Adversarial training can result in regularization; even further regularization than dropout.
- RBF networks are resistant to adversarial examples.

The new loss (or objective) function has a regularizer term denoted by the fast gradient sign method:

$$\tilde{J}(\boldsymbol{\theta}, \boldsymbol{x}, y) = \alpha J(\boldsymbol{\theta}, \boldsymbol{x}, y) + (1 - \alpha) J(\boldsymbol{\theta}, \boldsymbol{x} + \epsilon \operatorname{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$$

Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).

Generative Adversarial Networks (GANs)