Intriguing properties of neural networks

Szegedy, Zaremba, Sutskever, Bruna, Erhan, Goodfellow & Fergus.
ArXiv (2013).

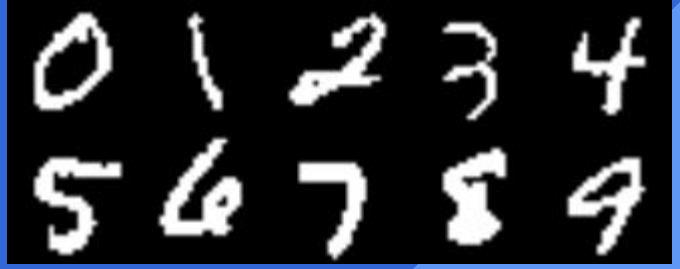
Presented by Juran Zhang

Two counter-intuitive properties of neural networks

1. There is no distinction between individual high level units and random linear combinations of high level units

2. Input and output mapping are discontinuous

Datasets

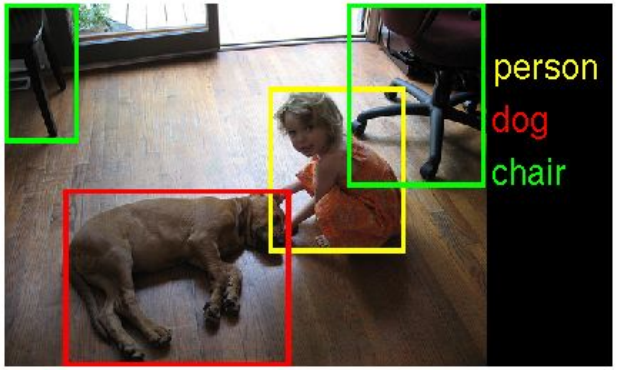


MNIST dataset

ImageNet

QuocNet:

- ~ 10M image samples from Youtube
- Unsupervised, ~1 billion learnable parameters



First property

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. **Rich feature hierarchies for accurate object detection and semantic segmentation**.

Matthew D Zeiler and Rob Fergus. **Visualizing and understanding convolutional neural networks**.

The inspection of individual units makes the implicit assumption that the units of the last feature layer have semantic information

First property

“It is the space, rather than the individual units, that contains the semantic information in the high layers of neural networks.”

(a) Unit sensitive to lower round stroke.

(b) Unit sensitive to upper round stroke, or lower straight stroke.

(c) Unit senstive to left, upper round stroke.

(d) Unit senstive to diagonal straight stroke.

Figure 1: An MNIST experiment. The figure shows images that maximize the activation of various units (maximum stimulation in the natural basis direction). Images within each row share semantic properties.

(a) Direction sensitive to upper straight stroke, or lower round stroke.

(b) Direction sensitive to lower left loop.

(c) Direction senstive to round top stroke.

(d) Direction sensitive to right, upper round stroke.

Figure 2: An MNIST experiment. The figure shows images that maximize the activations in a random direction (maximum stimulation in a random basis). Images within each row share semantic properties.

(a) Unit sensitive to white flowers.

(b) Unit sensitive to postures.

(c) Unit senstive to round, spiky flowers.

(d) Unit senstive to round green or yellow objects.

Figure 3: Experiment performed on ImageNet. Images stimulating single unit most (maximum stimulation in natural basis direction). Images within each row share many semantic properties.

(a) Direction sensitive to white, spread flowers.

(b) Direction sensitive to white dogs.

(c) Direction sensitive to spread shapes.

(d) Direction sensitive to dogs with brown heads.

Figure 4: Experiment performed on ImageNet. Images giving rise to maximum activations in a random direction (maximum stimulation in a random basis). Images within each row share many semantic properties.

First property: conclusion

A single neuron’s feature is no more interpretable as a meaningful feature than a random set of neurons

Second property

The deep neural networks learn input-output mappings that are fairly discontinuous.

It is possible to cause the network to misclassify an image by applying a certain hardly perceptible perturbation

“Adversarial examples”

(a)

(b)

Figure 6: Adversarial examples for QuocNet [10]. A binary car classifier was trained on top of the last layer features without fine-tuning. The examples on the left are recognized correctly as cars, while the images in the middle are not recognized. The rightmost column is the magnified absolute value of the difference between the two images.

(a) (b)

Figure 5: Adversarial examples generated for AlexNet [9].(Left) is correctly predicted sample, (center) difference between correct image, and image predicted incorrectly magnified by 10x (values shifted by 128 and clamped), (right) adversarial example. All images in the right column are predicted to be a "ostrich, Struthio camelus", which is fast-running African flightless bird with two-toed feet, largest living bird. Average distortion based on 64 examples is 0.006508.

Second property: generate “adversarial examples”

“Perturbations are found by optimizing the input to maximize the prediction error”

Pedro Tabacof and Eduardo Valle, **Exploring the Space of Adversarial Images**

https://arxiv.org/pdf/1510.05328v5.pdf

Second property: observation

Cross networks generalization:

Applies to all dataset

Cross model generalization:

Hyper-parameters

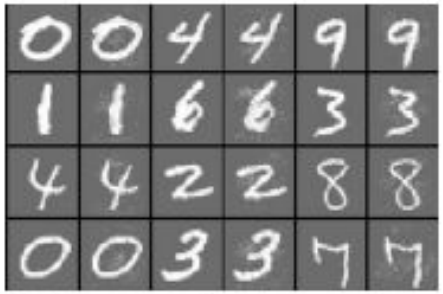
Cross training-set generalization:

Applies to different subsets

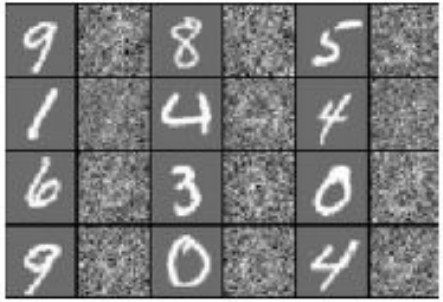
Second property: cross model generalization

Adversarial examples stay hard for different hyper-parameters.

(a) Even columns: adversarial examples for a linear (FC) classifier (stddev=0.06)



(b) Even columns: adversarial examples for a 200-200-10 sigmoid network (stddev=0.063)



(c) Randomly distorted samples by Gaussian noise with stddev=1. Accuracy: 51%.

Figure 7: Adversarial examples for MNIST compared with randomly distorted examples. Odd columns correspond to original images, and even columns correspond to distorted counterparts. The adversarial examples generated for the specific model have accuracy 0% for the respective model. Note that while the randomly distorted examples are hardly readable, still they are classified correctly in half of the cases, while the adversarial examples are never classified correctly.

	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

Table 2: Cross-model generalization of adversarial examples. The columns of the Tables show the error induced by distorted examples fed to the given model. The last column shows average distortion wrt. original training set.

Linear classifiers with various weight decay parameters lamda

All others: lossdecay = λ Σwi^2 /k, k is number of units in the layer

Second property: cross training set generalization

Adversarial examples stay hard for models trained on different subsets of a dataset.

	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%
Distorted for FC100-100-10 amplified to stddev=0.1	100%	98%	43%
Distorted for FC123-456-10 amplified to stddev=0.1	96%	100%	22%
Distorted for FC100-100-10' amplified to stddev=0.1	27%	50%	100%
Gaussian noise with stddev=0.1	2.6%	2.8%	2.7%

Table 4: Cross-training-set generalization error rate for the set of adversarial examples generated for different models. The error induced by a random distortion to the same examples is displayed in the last row.

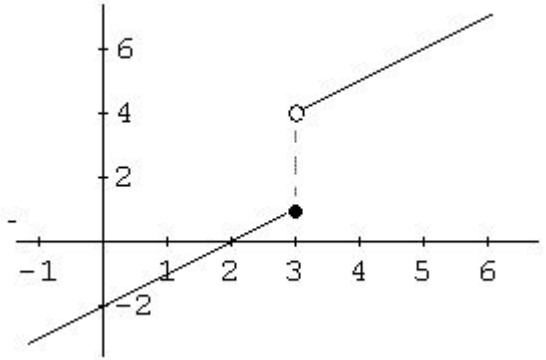
60000 MNIST partitioned into P1 and P2, trained on non-CNNs

Two networks on P1: cumulative effect of changing hyper-parameters and training sets at the same time

Magnify distortion to make stddev 0.1 before feeding back

Second property: main contribution

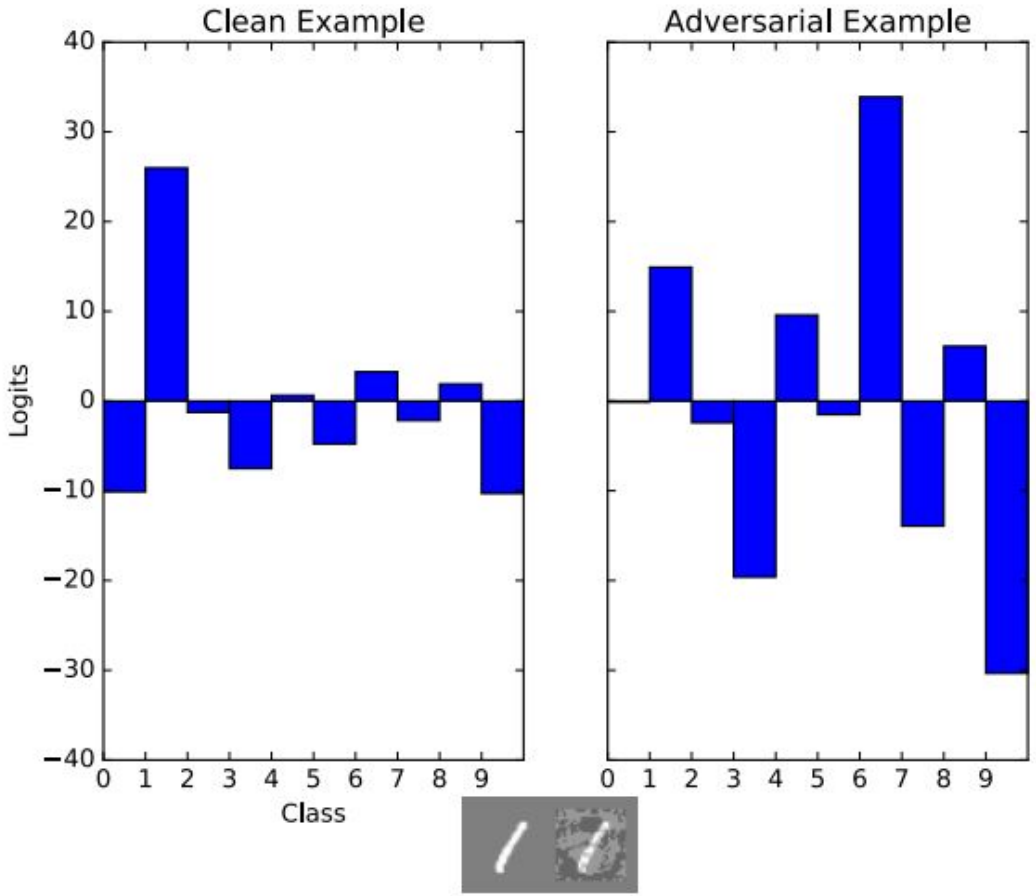
For deep neural networks, the smoothness assumption that **tiny perturbations of a given image do not normally change** the classification result **does not hold!**



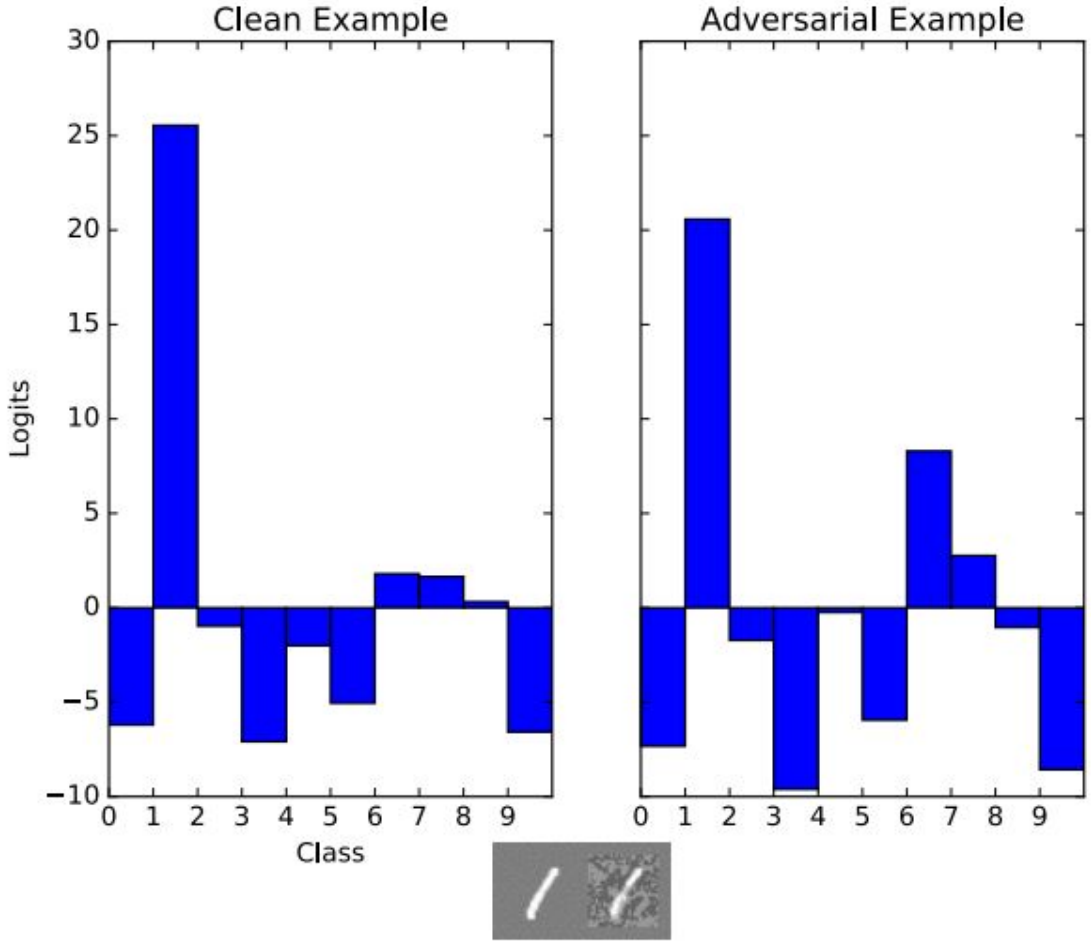
Second property: highlight

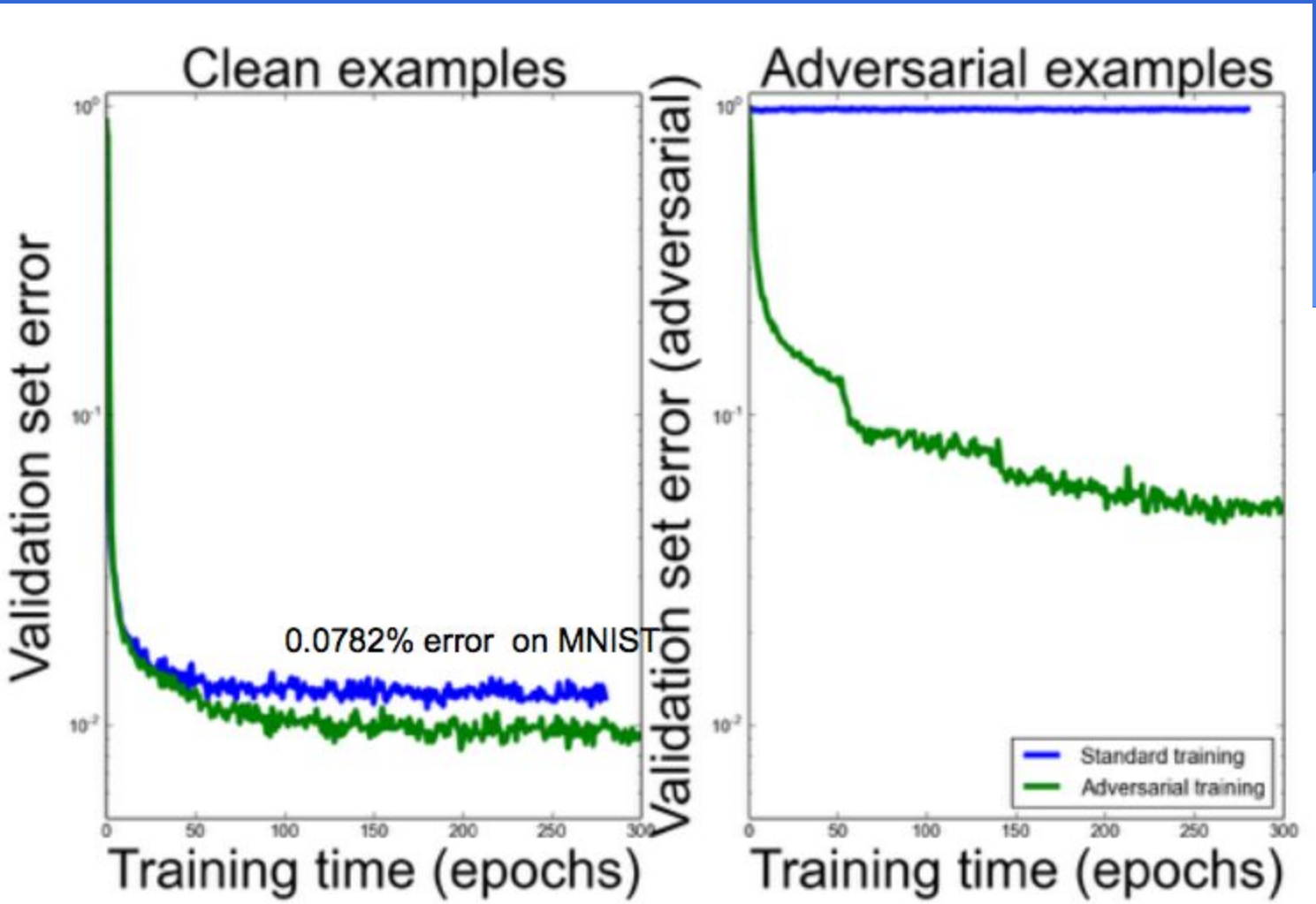
They **use adversarial examples to train** a non-CNN with test error below 1.2% (improved from 1.6% and 1.3%)

Pertubation's effect on class distributions



Pertubation's effect after adversarial training




Clean examples
Adversarial examples
Validation set error
Validation set error (adversarial)
0.0782% error on MNIST
Standard training
Adversarial training
Training time (epochs)
Training time (epochs)

Second property: fails to illustrate

1. Only experiment results on a subset of MNIST are presented.
2. Does not provide results on convolutional models.
3. Lacks a focus: write several papers

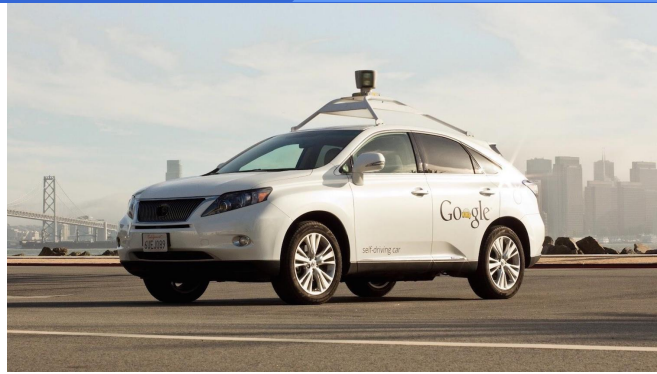
Second property: thoughts

Is this a critical problem for practical applications?

How often the adversarial examples appear?

Deep flaw or more cautious interpretation?

Security implications?



http://www.iro.umontreal.ca/~memisevr/dlss2015/goodfellow_adv.pdf