Adversarial Training

Sargur N. Srihari
srihari@cedar.buffalo.edu

Do nets have Human-level understanding?

- In many cases, neural networks have begun to reach human level performance when evaluated on an i.i.d. test set
  - Have they reached human level understanding?
- To probe the level of understanding we can probe examples that model misclassifies
  - Even neural networks that perform at human level accuracy have a 100% error rate on examples intentionally constructed!

Adversarial examples

- An optimization procedure is used to search for an input $\boldsymbol{x}$' near data point $\boldsymbol{x}$ such that the model output is very different at $\boldsymbol{x}$'
  - In many cases, $\boldsymbol{x}$' can be so similar to $\boldsymbol{x}$ that a human observer cannot tell the difference between the original example and the adversarial example
  - But the network makes a highly different prediction

Adversarial Example Generation

We add to $\boldsymbol{x}$ an imperceptibly small vector
Its elements are equal to the sign of the elements of the gradient of the cost function wrt the input. It changes Googlenet's classification of the image

$+ .007 \times$

$=$

$\boldsymbol{x}$

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

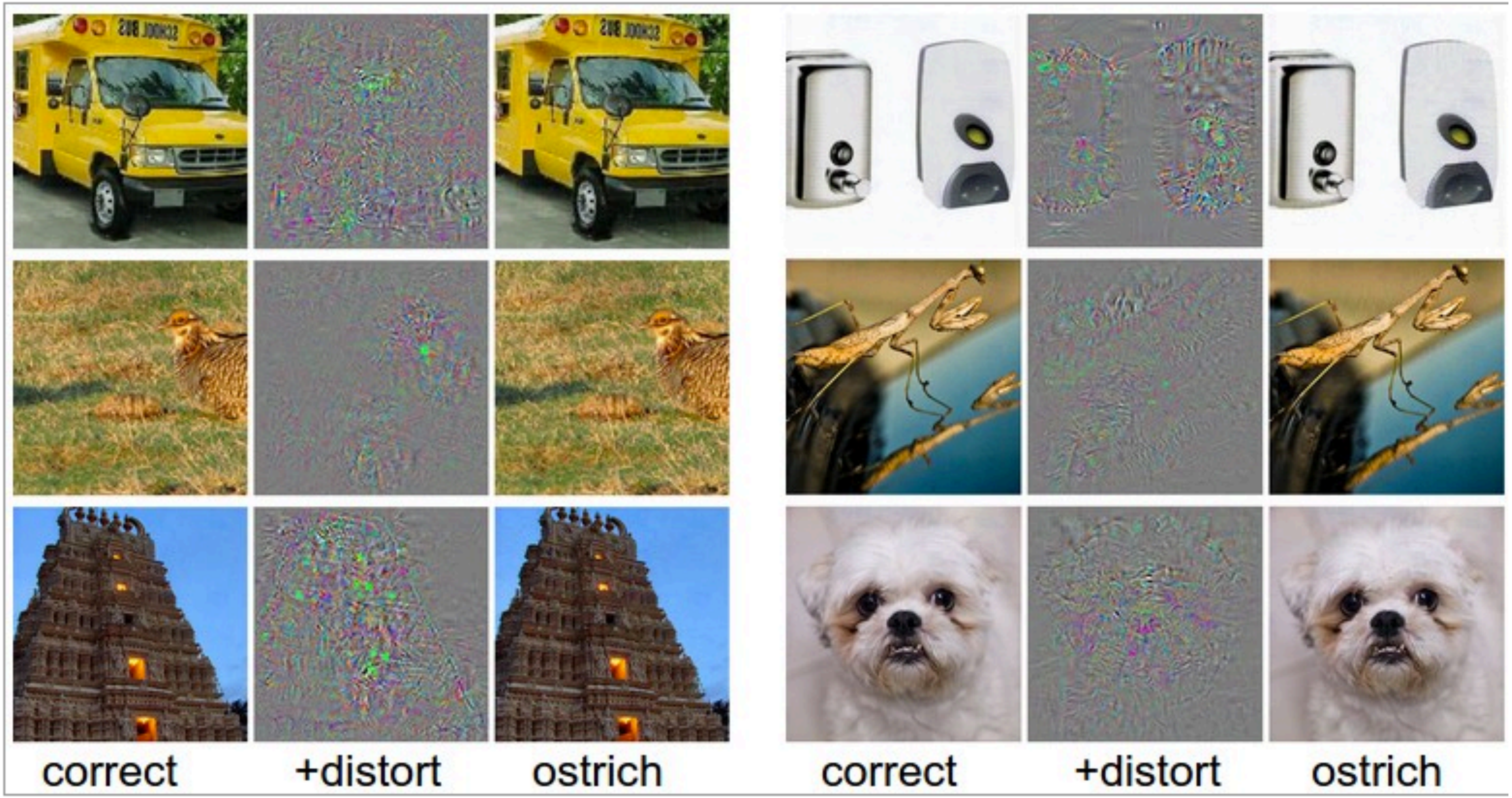
$\boldsymbol{x} + \epsilon\, \text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

y ="panda"
with 58% confidence

y ="nermatode"
With 8.2% confidence

y ="gibbon"
With 99% confidence

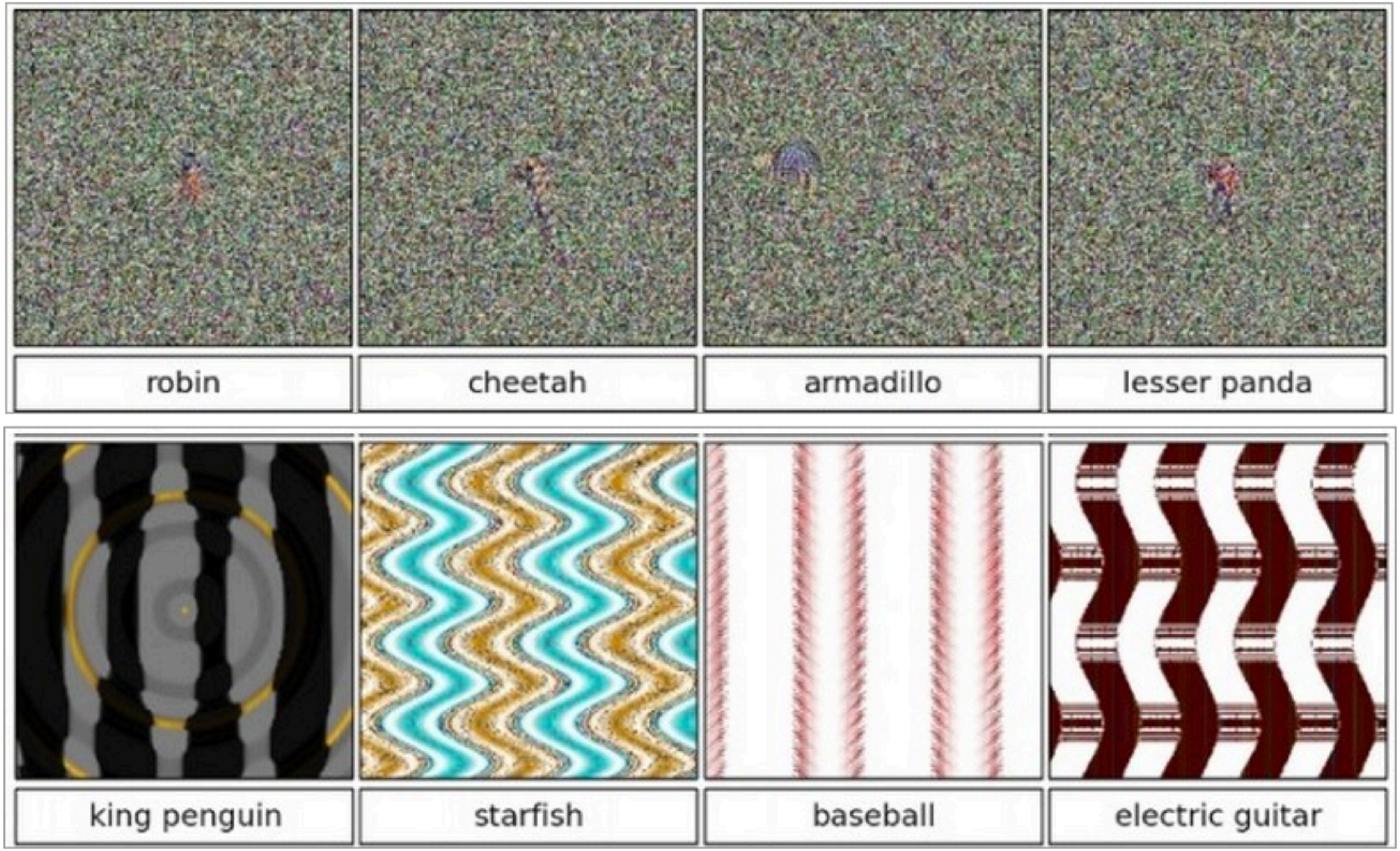
More examples



Take a correctly classified image (left image in both columns), and add a tiny distortion (middle) to fool the ConvNet with the resulting image (right).

http://karpathy.github.io/2015/03/30/breaking-convnets/

Some more examples



These images are classified with >99.6% confidence as the shown class by a Convolutional Network.

Uses of adversarial training

- Adversarial examples have many implications
  - E.g., they are useful in computer security
    - Adversarial examples are hard to defend against
  - They are interesting in the context of regularization
    - Using adversarially perturbed samples we can reduce error rate on test set

Cause of adversarial examples

- Primary cause is excessive linearity
  - Neural networks are built primarily out of linear building blocks
    - The overall function often proves to be linear
  - Linear functions are easy to optimize
  - But the value of a linear function can change rapidly with numerous inputs
  - If we change input by ε then a linear functions with weights $\boldsymbol{w}$ can change by $\varepsilon||\boldsymbol{w}||$ which can be very large in high-dimensional spaces

Adversarial Training

- Adversarial training discourages highly sensitive local behavior
- By encouraging network to be locally constant in the neighborhood of the training data
- This can be seen as a way of explicitly introducing a local constancy prior into supervised neural nets

Adversarial training and Capacity

- Adversarial training illustrates the power of using a large function family in combination with aggressive regularization
  - Purely linear models, like logistic regression, are unable to resist adversarial examples because they are forced to be linear
- Neural networks are able to represent functions that can range from nearly linear to nearly locally constant
  - Thus can capture linear trends as well as learning to resist local perturbation

Relation to Semi-supervised Learning

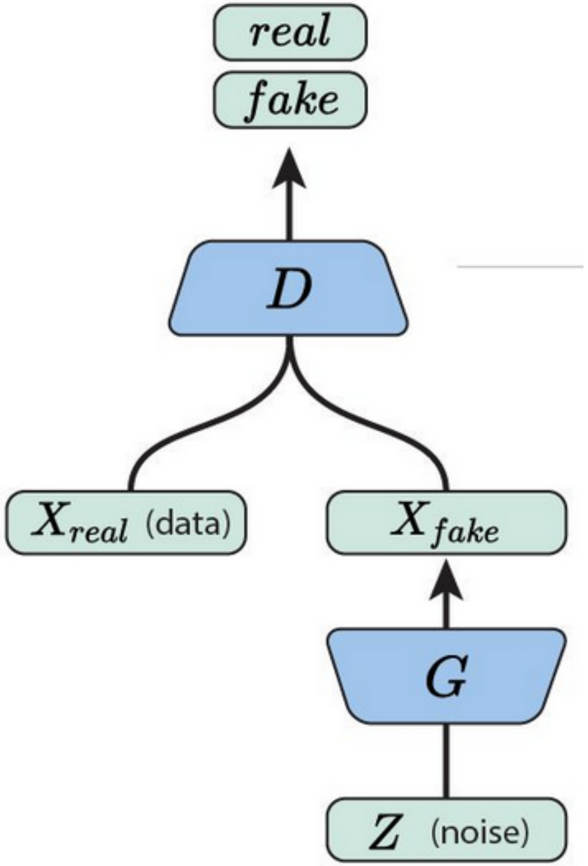
- Adversarial examples provide a means of accomplishing semi-supervised learning
- At a point $\boldsymbol{x}$ that is not associated with a label in a dataset, the model itself assigns some label $\hat{y}$
- It may not be the true label, but if model is of high quality then $\hat{y}$ has a probability of being the true label
- We can seek an adversarial example $\boldsymbol{x}'$ that causes the classifier to output a label y' with with $y' \neq \hat{y}$

Virtual Adversarial Examples

- Adversarial examples generated with using not the true label but a label provided by a trained model are called *Virtual Adversarial Examples*
  - The classifier may then be trained to assign the same label to $\boldsymbol{x}$ and $\boldsymbol{x}$'
  - This encourages the classifier to learn a function that is robust to small changes anywhere along the manifold where the unlabeled data lie
- Assumption motivating this approach
  - different classes lie on disconnected manifolds
    - A small perturbation should not be able to jump from one class manifold to another class manifold

Generative Adversarial Network

- GANs are a way to make a generative model by having two neural networks compete with each other



The discriminator tries to distinguish genuine data from forgeries created by the generator

The generator turns random noise into imitations of the data, in an attempt to fool the discriminator