Explaining and Harnessing Adversarial Examples

I.Goodfellow, J.Shlens and C.Szegedy

Minji Kim, Seong Jin Lee

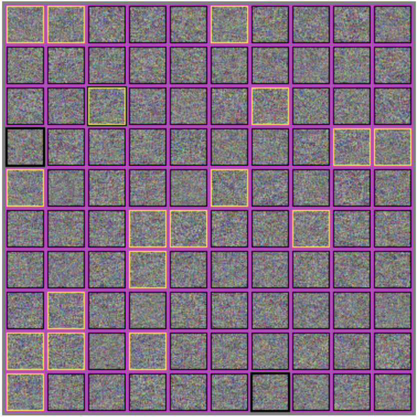
Statistics and Operations Research
UNC at Chapel Hill

April 20, 2022

Section 1

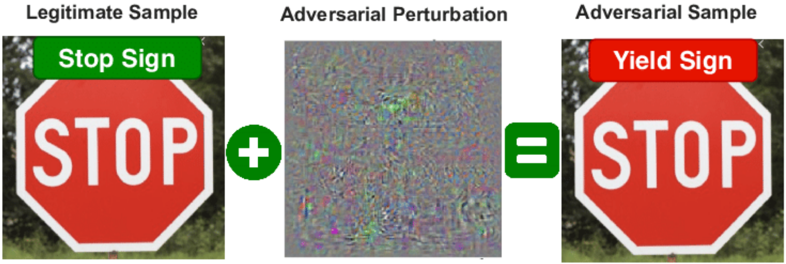
Introduction

Deep Learning Isn't Perfect



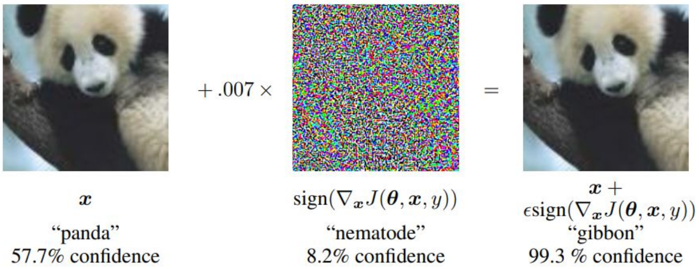
- Models that perform well on test dataset classify these random noises as certain objects with high confidence.
- There are fundamental blind spots in the training algorithm.

Adversarial Examples



- Models sometimes misclassify when small noise is added.
- Consider a self-driving car that uses NN to read traffic signs.
- We need models that are robust to these adversarial 'attacks'.

Adversarial Examples


$+ .007 \times$
$=$
$\boldsymbol{x}$
"panda"
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

- To be robust to adversarial attacks, we must first identify these 'attacks'.
- Why is this happening? How can we make such examples?

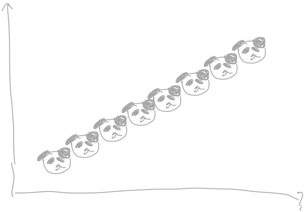
Paper Overview

Previous studies have explained this due to extreme nonlinearity and overfitting of neural networks. This research suggests . . .

1. **Linear behavior in high-dimensional spaces** is sufficient to cause adversarial examples.
2. Based on this view, design FGSM: a fast method of generating adversarial examples that makes adversarial training practical.
3. Adversarial training provides an additional regularization benefit beyond that provided by using dropout.

Section 2

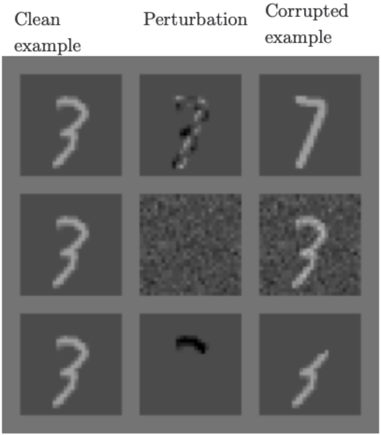
Linear Explanation of Adversarial Examples



Max Norm Bounded Noise

- Consider the 8 bit image data, where information is contained in the range $\{0/256, 1/256, 2/256, \ldots 255/256\}$.
- Due to limitation of precision, changes below $1/256$ is discarded. In other words, noises with values less than $1/256$ in each pixel are ignored.
- Any noise with maximum norm bounded by $1/256$ should not change the output.

Max Norm Bounded Noise



Perturbation changes the true class

Random perturbation does not change the class

Perturbation changes the input to “rubbish class”

All three perturbations have L2 norm 3.96

This is actually small. We typically use 7!

Linear Explanation

- Consider an input x and perturbed input $\tilde{x} = x + \eta$. With weight vector w we have

$$w^\top \tilde{x} = w^\top x + w^\top \eta$$

- We want to maximize the difference between the original output and the perturbed output, in other words solve

$$\underset{\eta}{\text{maximize}}\ w^\top \eta \text{ subject to } \|\eta\|_\infty \leq \epsilon$$

Linear Explanation

- The optimization problem:

$$\underset{\eta}{\text{maximize}}\ w^\top \eta \text{ subject to } \|\eta\|_\infty \le \epsilon$$

- Suppose $w = (2, -1, 3)^\top$. The problem becomes

$$\underset{\eta}{\text{maximize}}\ 2\eta_1 - \eta_2 + \eta_3 \text{ subject to } |\eta_i| \le \epsilon$$

  which has the solution $\eta = (+\epsilon, -\epsilon, +\epsilon)^\top$

- The solution to the original problem is $\eta = \epsilon \cdot \text{sign}(w)$.

Linear Explanation

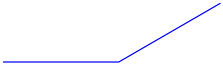
- Now we have the optimal perturbation $\eta = \epsilon \cdot \text{sign}(w)$. Note that $w^\top \eta = \epsilon \|w\|_1$.
- While the input is perturbed only by $\|\eta\|_\infty = \epsilon$, the activation differs by $w^\top \eta = \|w\|_1 = \epsilon mn$, where m is the average magnitude of the weight.
- With small ϵ max-norm bound, input x and $\tilde{x}$ are almost same, but the output $w^\top(\tilde{x} - x)$ may change a lot if n large enough.
- Even simple linear models can have adversarial examples with sufficient dimensionality.

Section 3

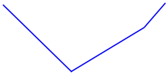
Fast Generation of Adversarial Examples

Neural Networks are Linear

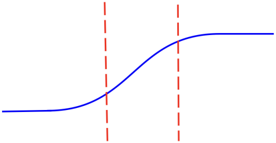
Rectified linear unit



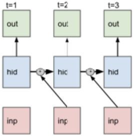
Maxout



Carefully tuned sigmoid



LSTM



- Adversarial examples can be explained as a property of high-dimensional dot products. They are a result of models being too linear, rather than too nonlinear.

Fast Gradient Sign Method

- Let θ be the parameters, x be the input , y be the output and $J(\theta, x, y)$ be the associated cost.
- Now consider the perturbed input $\tilde{x} = x + \eta$. With the maximum norm bound we want to solve

$$\underset{\eta}{\text{maximize}}\ J(\theta, \tilde{x}, y), \qquad \text{subject to } ||\eta||_\infty \leq \epsilon$$

- Using linear approximation of J, we have

$$\begin{aligned} J(\theta, \tilde{x}, y) &= J(\theta, x, y) + (\tilde{x} - x)^\top \nabla_x J(x) \\ &= J(\theta, x, y) + \eta^\top \nabla_x J(x) \end{aligned}$$

Fast Gradient Sign Method

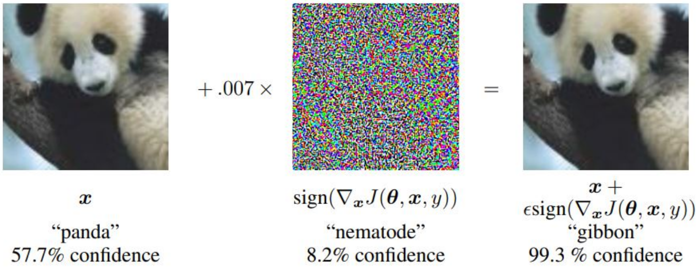
- Therefore we need to solve

$$\underset{\eta}{\text{maximize}}\ \eta^\top \nabla_x J(x) \quad \text{subject to } \|\eta\|_\infty \leq \epsilon$$

- The "fast gradient sign method" creates an attack $\tilde{x} = x + \eta$ using the optimal ϵ-max-norm constrained perturbation of

$$\eta = \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y))$$

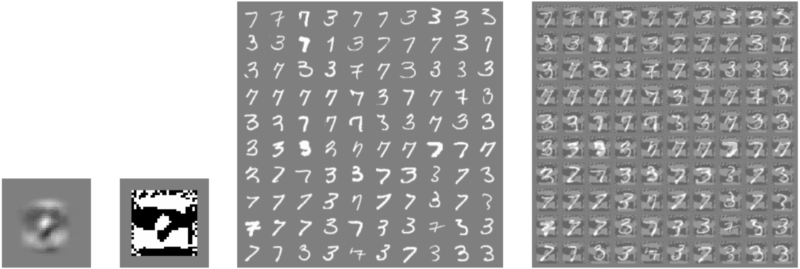
FGSM: Example



$+ .007 \times$
$=$
$\boldsymbol{x}$
"panda"
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

- This simple, cheap algorithm adding an imperceptibly small vector based on the sign of the gradient of the cost function generates misclassified examples.

FGSM: Example



(a) (b) (c) (d)

- FGSM applied to logistic regression:
  (a) weights trained (b) sign of weights
  (c) 1.6% error rate for the original model
  (d) 99% error rate for the adversarial examples

Section 4

Adversarial Training

Adversarial Training

- Train the model with the objective function

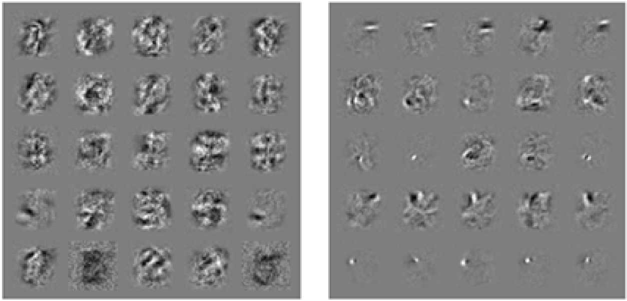
$$\tilde{J}(\theta, x, y) = \alpha J(\theta, x, y) + (1-\alpha) J(\theta, x + \epsilon \text{sign}(\nabla_x J(\theta, x, y)), y)$$

- Two benefits are (i) additional regularization and (ii) more robustness on adversarial examples.
- The loss can be interpreted as minimizing the worst case error when the data is perturbed. It is different from just adding noises with ϵ-max norm.
- The procedure can be viewed as a form of active learning, where the model labels nearby points by itself without human labeler.

Results of Adversarial Training

- Training on a mixture of adversarial and clean examples reaped additional regularizational benefit on normal test errors.
  - Training a maxout network regularized with dropout reduced the error rate from 0.94% to 0.84%.
- Moreover, adversarially trained model showed greater robustness on adversarial examples.
  - Error rate on adversarial examples based on the FGSM dropped from 89.4% to 17.9% with adversarial training.

Observation



- Weights of the adversarially trained model (right) are significantly more localized and interpretable than before (left).

Limitations

Inside:

- The adversarial training result reduced an error rate, but when the model misclassify an adversarial example, it still predicts wrong with high confidence.

Outside:

- The method relied on a linear approximation of the loss function around data points.
- The models are robust against the FGSM adversarial examples, but vulnerable to other attacks such as those generated by iterative PGD method introduced later.
- In general, FGSM adversary produces restricted set of adversarial examples where the trained networks overfit and they did not exhibit robustness on large ϵ adversarial examples.

Section 5

Further Discussion

Model Capacity

- NN's are fooled by adversarial examples with high confidence.
- The RBF networks uses the non linear function

$$p(y = 1|x) = \exp((x - \mu)^\top \beta(x - \mu))$$

- Shallow RBF network gets 55% error rate on adversarial examples by FGSM, but its confidence on mistaken examples is only 1.2% where its average confidence on clean samples is 60.6%.
- Though RBF has low capacity, but it respond correctly by reducing its confidence on points it does not "understand".

Alternative Approach : Weight Decay

- Consider the logistic regression

$$\mathbb{E}_{x,y}\zeta(-y(w^\top x + b))$$

- We want our model to minimize

$$\mathbb{E} \max_{\|\eta\|_\infty < \epsilon} \zeta(-y(w^\top (x+\eta) + b))$$

- If we consider the worst case with perturbation $\epsilon \cdot \text{sign}(w)$, since $w^\top \text{sign}(w) = \|w\|_1$,

$$\mathbb{E}_{x,y}\zeta(y(\epsilon \cdot \|w\|_1 - w^\top x - b))$$

- This is somewhat similar to the L^1 regularization but has limitations.

Projected Gradient Descent

- We want to minimize $\rho(\theta)$ where

$$\rho(\theta) = \mathbb{E}_{(x,y)\sim D} \max_{\eta \in \mathcal{S}} L(\theta, x + \eta, y)$$

  where $\mathcal{S}$ is the ϵ-max norm ball.

- In FGSM we use the linear approximation and use

$$x + \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y))$$

  Instead PGD uses the multi-step variant and use

$$x^{t+1} = \Pi_{x+\mathcal{S}}(x^t + \alpha \cdot \text{sign}(\nabla_x L(\theta, x^t, y)))$$

Generalization of Adversarial Examples

- Adversarial examples ”generalize” as a adversarial example generated from one model is adversarial on a different model.
- The authors hypothesize that neural networks all resemble a global linear classifier trained on the same training set.

Alternative Hypothesis

- Generative training could provide more constraint or cause the model to distinguish real from fake
  - Generative model is still vulnerable to adversarial examples, reported 97.5% error rate using the MNIST data set.
- Averaging over many models can cause adversarial examples (AE) to wash out
  - Ensemble of 12 maxout networks on MNIST showed 91.1% error rate on AE designed to perturb the entire ensemble, while 87.9% error was reported using AE designed to perturb a particular member.
  - Ensemble provides limited resistance to adversarial perturbation.

Reference

- Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
- Carlini, N., & Wagner, D. (2017, May). Towards evaluating the robustness of neural networks. In 2017 ieee symposium on security and privacy (sp) (pp. 39-57). IEEE.
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.