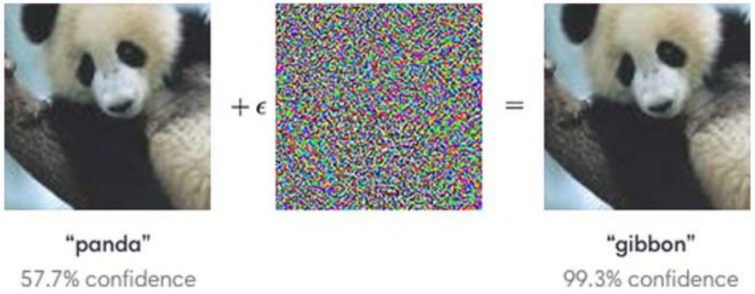
Ensemble Method as a Defense Against Adversarial Examples

By: Cale Harms, Colton Harper,
Krishnamohan Sunkara

Introduction



- Power of deep learning algorithms
- Examples: Autonomous vehicles, image classification, security
- Deep neural networks are vulnerable to small perturbations to images, resulting in a significant decrease in performance.

Motivation

TOM SIMONITE BUSINESS 03.09.18 07:00 AM

AI HAS A HALLUCINATION PROBLEM THAT'S PROVING TOUGH TO FIX

MAI SCHOTZ

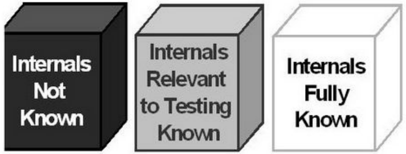
TECH COMPANIES ARE rushing to infuse everything with artificial intelligence, driven by big leaps in the power of machine learning software. But the deep-neural-network software fueling the excitement has a troubling weakness: Making subtle changes to images, text, or audio can fool these systems into perceiving things that aren't there.

That could be a big problem for products dependent on machine learning, particularly for vision, such as self-driving cars. Leading researchers are trying to develop defenses against such attacks—but that's proving to be a challenge.

Outline

- Introduction
- Outline
- Generating Adversarial Examples
- Defense Strategies
- Ensemble Method
- Results
- Conclusions
- Future Works

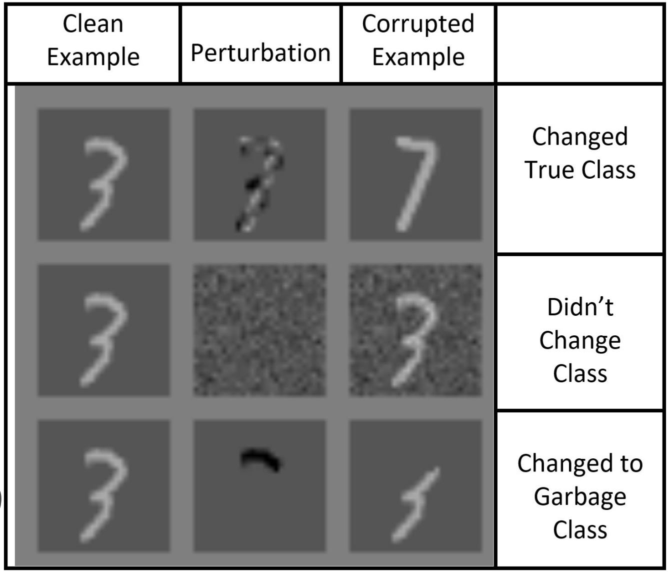
Generating Adversarial Examples



- Types of adversarial attacks
  - Targeted - an attack intentionally trying to perturb images to a specific class when misclassified.
  - Non-targeted - an attack to simply have images misclassified.
  - White Box - an attack that uses the specifications of a model to generate adversarial examples.
  - Black Box - Only inputs and outputs are known to generate an adversarial example.

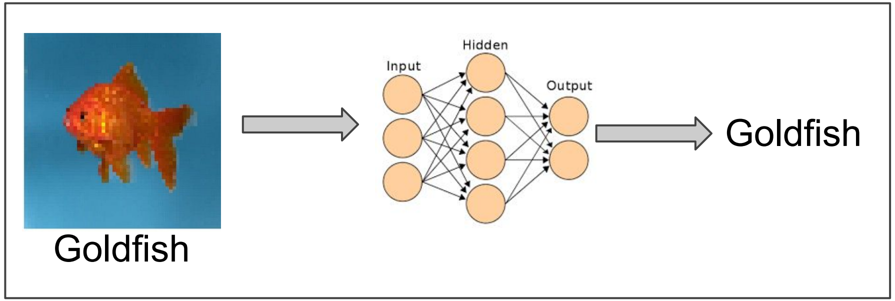
Adversarial Attacks

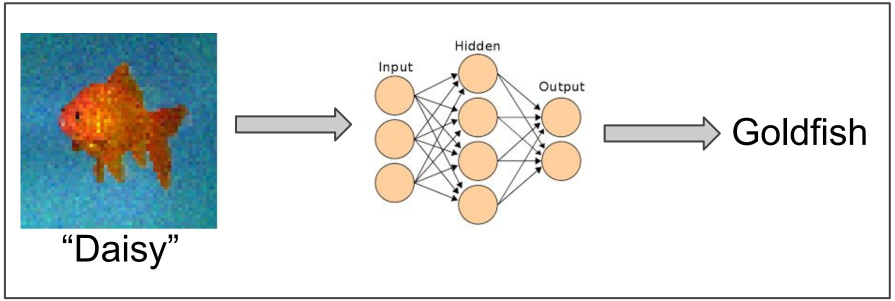
- Fast Gradient Sign Method (FGSM)
  - $x_{FGSM} = x + \varepsilon \cdot sign[\nabla_x J(\Theta, x, y)]$
- Basic Iterative Method (BIM)
  - $x_0 = x,$
  - $x_i = clip_{x,\varepsilon}(x_{i-1} + \alpha\, sign[\nabla_{i-1} J(\Theta, x_{i-1}, y)])$
  - $x_{BIM} = \mathrm{x}_\mathrm{n}$
- Limited Memory BFGS (L-BFGS)
  - $minimize\ c \cdot ||x - x'||^2_2$
  - $such\ that\ x' \in [0,1]^n$



Defense Strategies: Adversarial Training

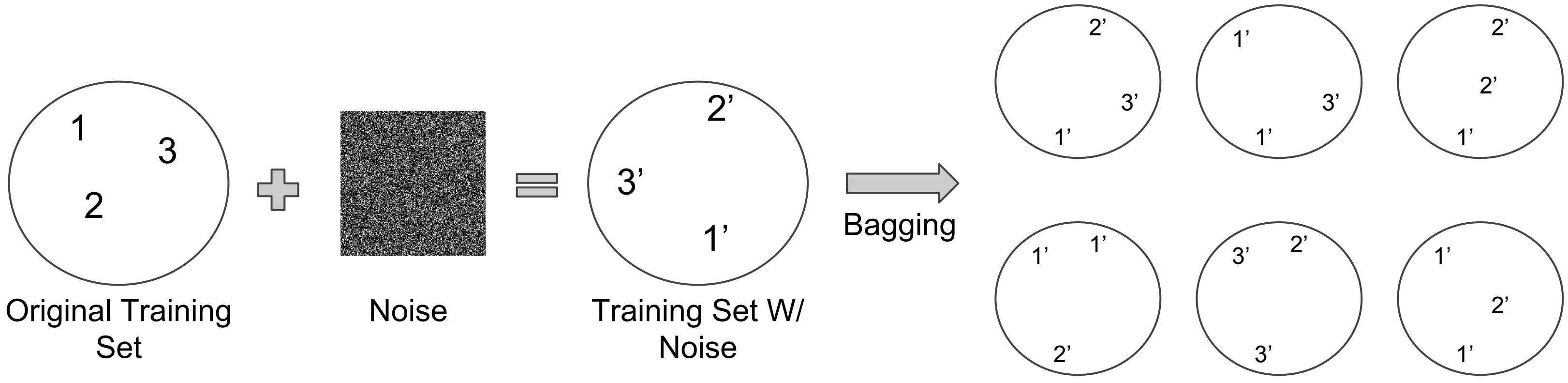
- Model trains on an image from the training set
- Generates adversarial example
- Trains on correctly labeled adversarial example
- Repeat





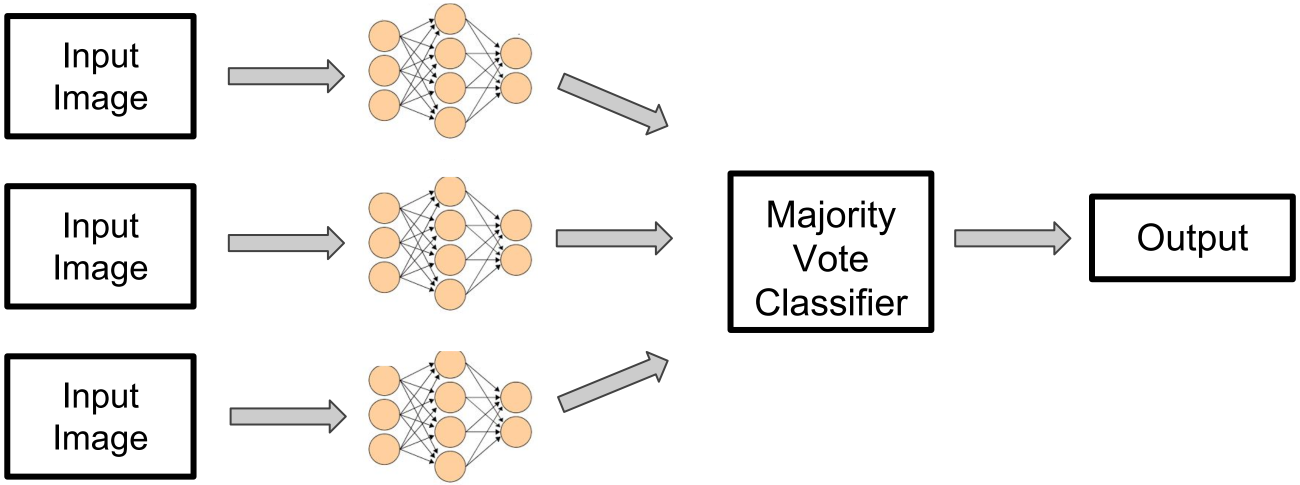
Defense Strategies: Bagging + Noise

- Given training data *T* with *m* data points
- Draw *m* samples (w/ replacement) from *T*
- Ensemble



Defense Strategies: Ensemble

- Set of classifiers
- Majority vote of classifiers



Implementation

- Datasets:
  - MNIST, CIFAR-10 & Tiny-ImageNet
- MNIST Model, CIFAR-10 Model, Inception V3
- Majority Vote Ensemble
- Generating Adversarial Attacks:
  - Cleverhans - A Python library using TensorFlow

Ensemble Implementation

- MNIST Architecture:

Layer Type	Parameters
2D Convolution Layer	32 filters
2D Convolution Layer	64 filters
Max Pooling	(2,2)
Dropout	0.25
Flatten	
Dense Layer	128 units
Dropout	0.5
Dense	10 units

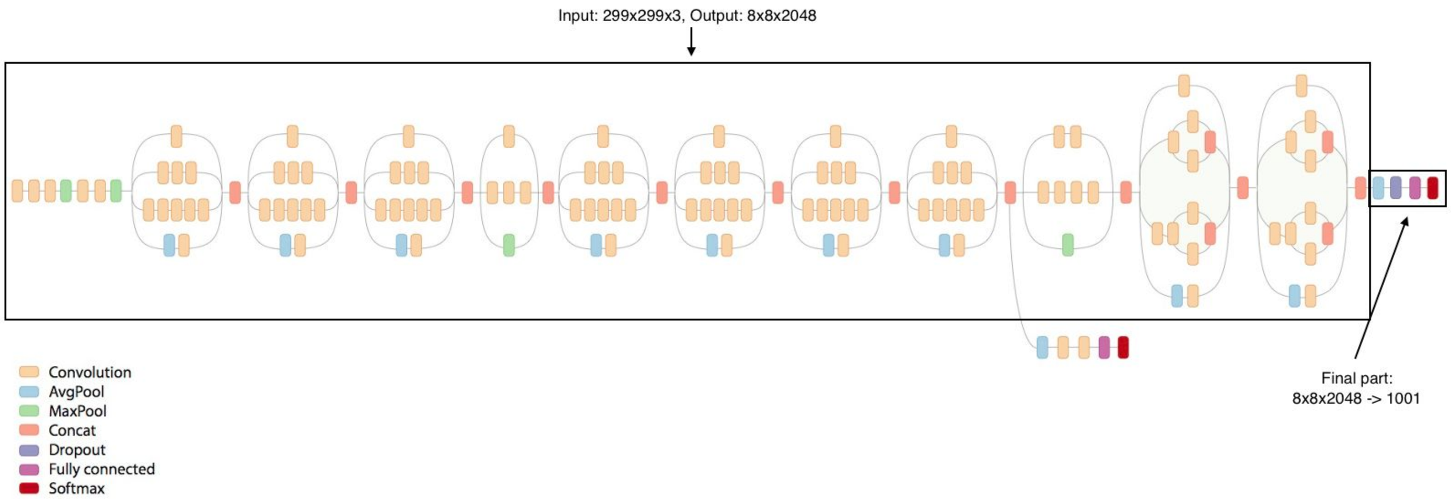
Implementation Continued

- CIFAR-10 Architecture:

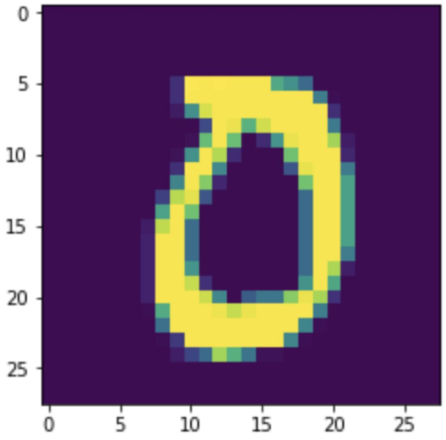
Layer Type	Parameters
2D Convolution Layer	32 filters
2D Convolution Layer	32 filters
Max Pooling	(2,2)
Dropout	0.25
2D Convolution Layer	64 filters
2D Convolution Layer	64 filters
Max Pooling	(2,2)
Dropout	0.25
Flatten	
Dense Layer	512 units
Dropout	0.5
Dense Layer	10 units

Implementation Continued

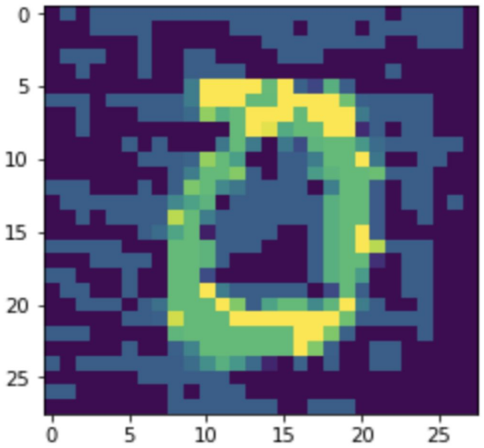
- Inception Architecture



Results

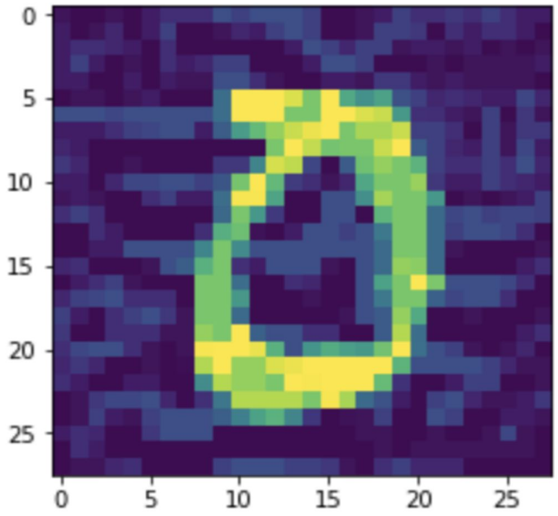


Original Image, Prediction:0

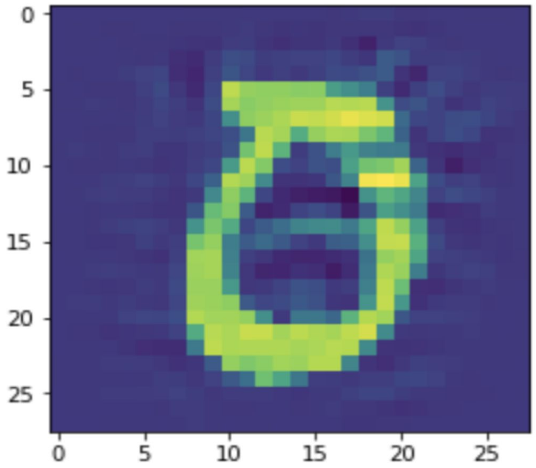


FGSM, Prediction: 3

Results Continued



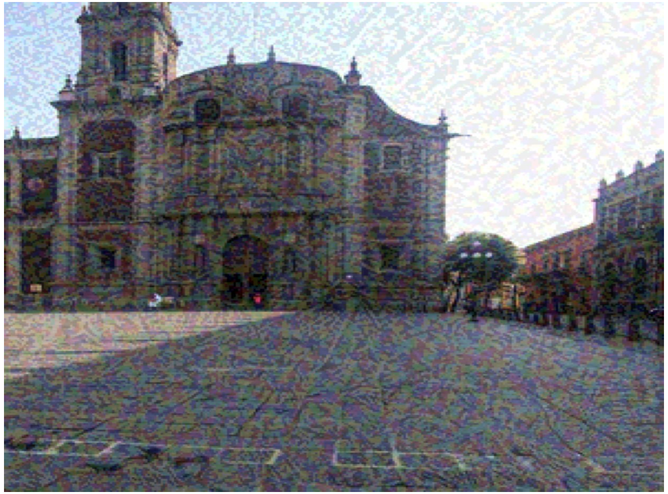
BIM, Prediction: 2



LBFGS, Target:6, Prediction:6

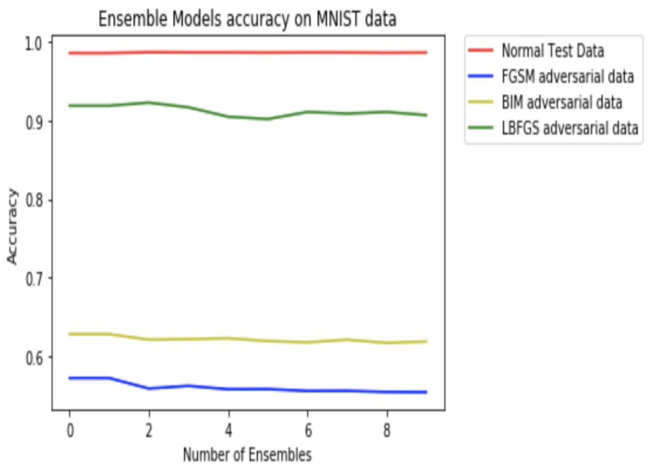
Results Continued

Original Image, Prediction:Building

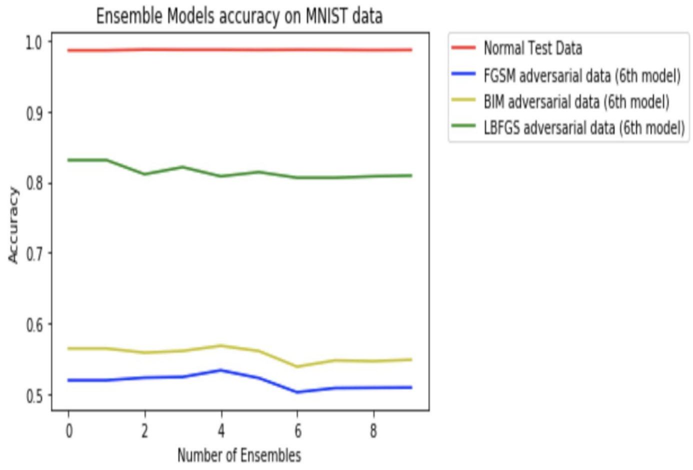


FGSM, Prediction:Bookcase

MNIST Results

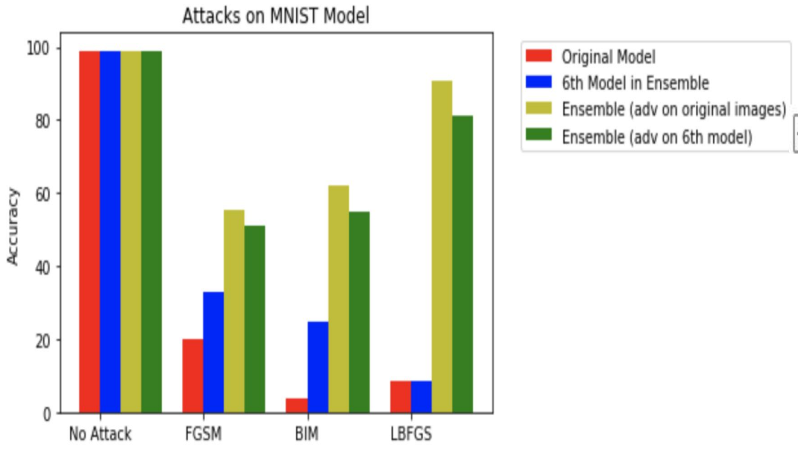


Ensemble models accuracy



Ensemble models accuracy on adversarial images of 6th model.

MNIST Results Continued

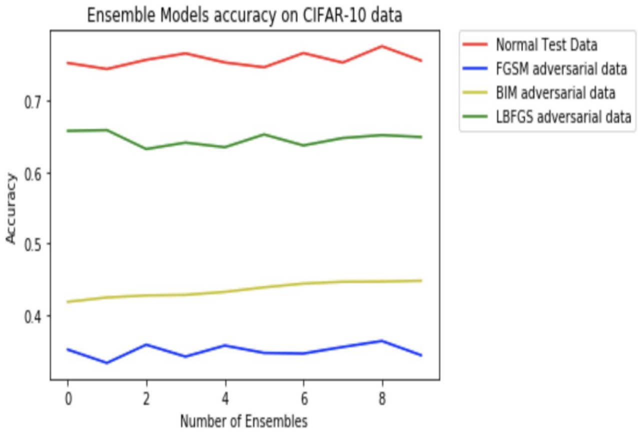


Accuracy on different attacks

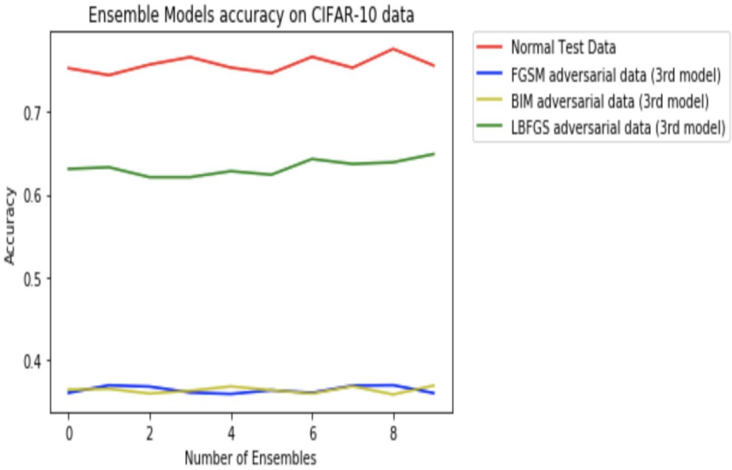
Adversarial Retraining	FGSM	BIM
Normal data	98.36	98.86
Adversarial data	96.68	93.16

Accuracy after adversarial training

CIFAR-10 Results

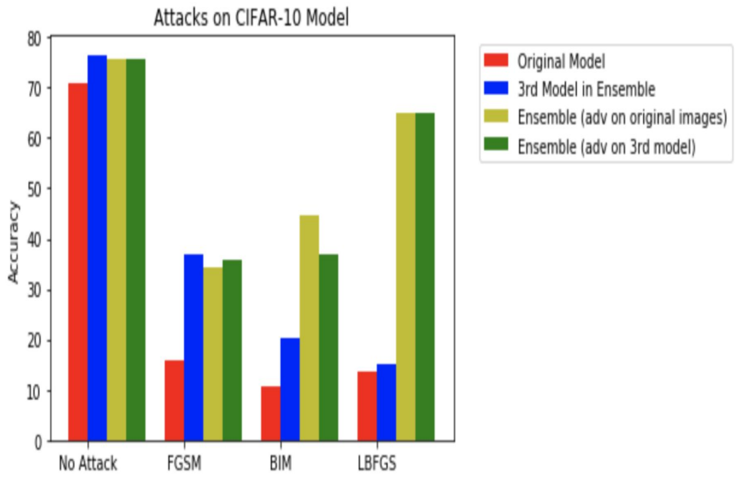


Ensemble models accuracy



Ensemble models accuracy on adversarial images of 3rd model.

CIFAR-10 Results Continued



Accuracy on different attacks

Adversarial Retraining	FGSM	BIM
Normal data	74.56	73.65
Adversarial data	66.78	58.97

Accuracy after adversarial training

Conclusions

- DNN's are highly vulnerable to adversarial examples
- Defending against adversarial examples is challenging
- Adversarial training makes the network more robust and is better compared to Ensemble methods.
- Ensemble methods like bagging + noise can provide
  - Increased accuracies on test data
  - Increase classifier robustness on these attacks

Future Work

- Successfully train the Inception model on HCC.
- Test on a wide variety of models and datasets.
- Combine Adversarial training with an ensemble.
- Test our network on different attacks.
- Apply Boosting on the ensemble.
- Test the L-BFGS attack on different target values.

References

[1] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, Dec. 2014.

[2] T. Strauss, M. Hanselmann, A. Junginger, H. Ulmer, Ensemble Methods as a Defense to Adversarial Perturbations Against Deep Neural Networks, arXiv preprint arXiv:1709.03423, 2017.

Thanks for your attention!

Cale Harms, Colton Harper, Krishna Sunkara
Department of Computer Science and Engineering
University of Nebraska-Lincoln
CSCE 496/896 Deep Learning