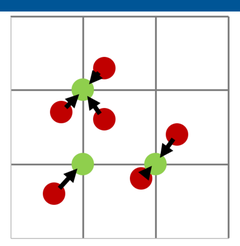
Feature Squeezing:
Detecting Adversarial Examples in Deep Neural Networks

[To appear in NDSS 2018]

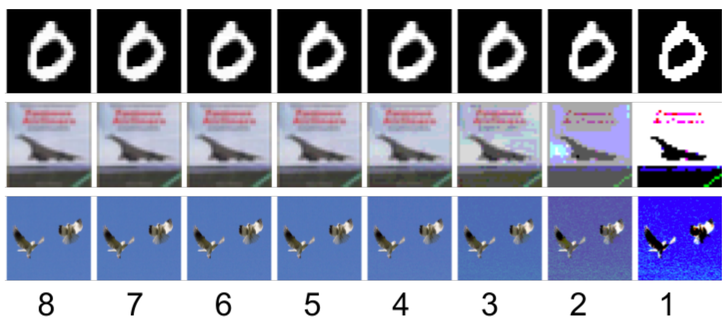
Weilin Xu, David Evans, Yanjun Qi
University of Virginia

• Motivation



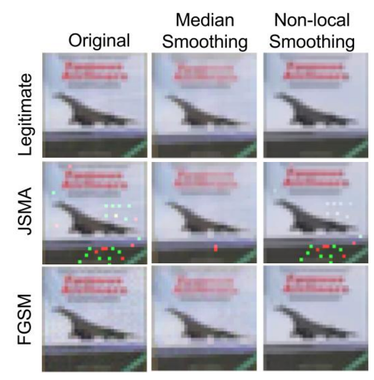
- Irrelevant features used in classification tasks are the root cause of adversarial examples.
- The feature spaces are unnecessarily too large in deep learning tasks: e.g. raw image pixels.
- We may reduce the search space of possible perturbations available to an adversary using *Feature Squeezing*.

• Feature Squeezing on Images



Bit Depth Reduction

- The 1-bit monochrome is not that different from 8-bit grayscale on MNIST, though the feature space is 128x smaller.
- Reducing to 4-bit per channel looks fine for color images, and the space is 4096x smaller.

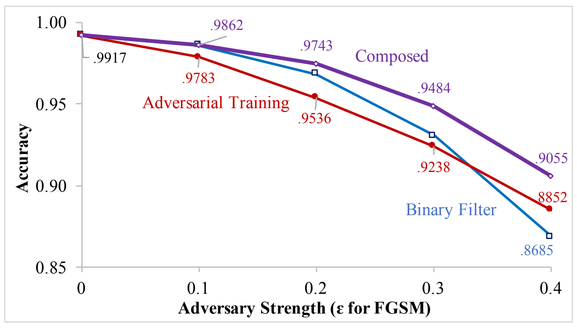


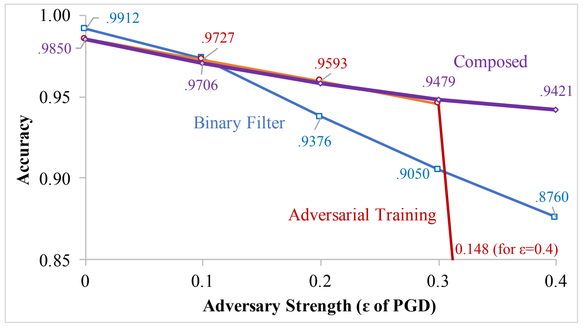
Spatial Smoothing

- The pixels are not totally independent for natural images.
- The smooth assumption could be used to filter the images.

• Combining with Adversarial Training

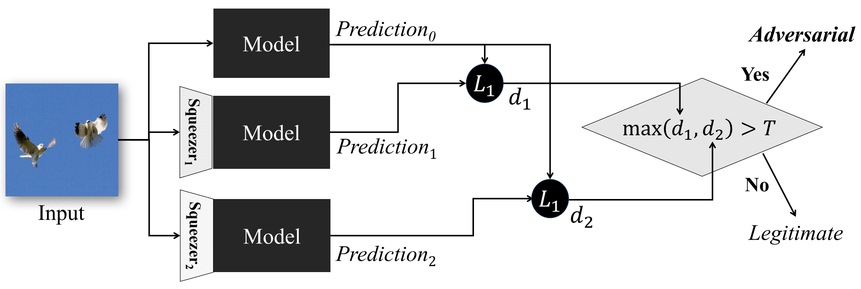
Since our approach modifies inputs rather than the model, it can easily be composed with any defense technique that operates on the model, such as adversarial training.





The composed one often produces the highest accuracy on MNIST, for both the FGSM-based and PGD-based adversarial training.

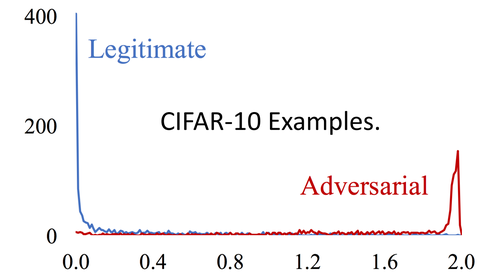
• Detecting Adversarial Examples



$$score^{(\mathbf{x},\mathbf{x}_{squeezed})} = \|g(\mathbf{x}) - g(\mathbf{x}_{squeezed})\|_1$$

$$score^{joint} = \max\left(score^{(\mathbf{x},\mathbf{x}_{sq1})}, score^{(\mathbf{x},\mathbf{x}_{sq2})}, \dots\right)$$

- Compare the model's prediction on the original sample with the same model's prediction on the sample after squeezing.
- The model's predictions for a legitimate example and its squeezed version should be similar.
- On the contrary, if the original and squeezed examples result in dramatically different predictions, the input is likely to be adversarial.

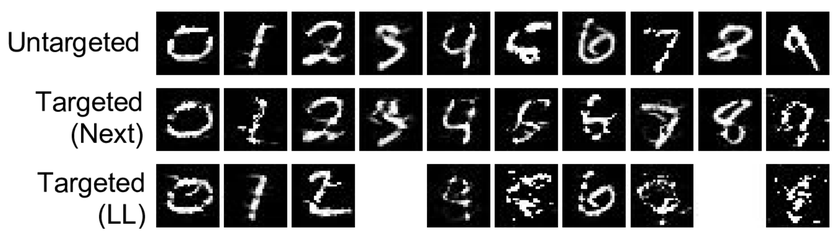


	Detecting SAEs*	ROC-AUC (Excluding FAEs)
MNIST	98.15%	99.44%
CIFAR-10	84.53%	95.74%
ImageNet	85.94%	94.24%

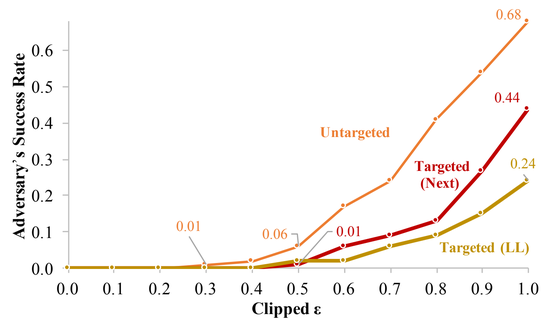
* False positive rates ~5%

• Adaptive Adversary

- Add one more term in the optimization objective of the CW_2 attack: [He et al. 2017]
  [Misclassification, Distance**, Detection Score**]
- Restart the algorithm with random initialization for non-differentiable components.
- Low success rates if we limit the perturbation magnitude.



Adaptive Adversarial Examples.



• Conclusion

- *Feature Squeezing* is effective against static adversary, though it is simple and inexpensive.
- *Feature Squeezing* could be used in many domains where deep learning is used, such as voice recognition.
- *Feature Squeezing* is not immune to adaptive adversary, but it substantially changes the challenge an adversary faces.
- Reproduce our results and compare with other works with EvadeML-Zoo: https://EvadeML.org/zoo