A Two-Pronged Defense against Adversarial Examples

Dongyu Meng
ShanghaiTech University, China

Hao Chen
University of California, Davis, USA
Neural networks in real-life applications

User authentication

Autonomous vehicle
Neural networks as classifier

Input → Classifier → Output (distribution)

- Panda 0.62
- Tiger 0.03
- Gibbon 0.11
Adversarial examples

Examples carefully crafted to
- look like normal examples
- cause misclassification

\[ p(x \text{ is panda}) = 0.58 \]
\[ p(x \text{ is gibbon}) = 0.99 \]

Attacks

$x' = x + \epsilon \cdot \text{sign}(\nabla_x \text{Loss}(x, l_x))$

Fast gradient sign method (FGSM)  
[Goodfellow, 2015]

Carlini’s attack  
[Carlini, 2017]

Iterative gradient  
[Kurakin, 2016]

Deepfool  
[Moosavi-Dezfooli, 2015]

......
## Defenses

<table>
<thead>
<tr>
<th>Method</th>
<th>Target Specific Attack</th>
<th>Modify Classifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adversarial training</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>[Goodfellow, 2015]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Defensive distillation</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>[Papernot, 2016]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detecting specific attacks</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>[Metzen, 2017]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>…….</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Desirable properties

Does not modify target classifier.

- Can be deployed more easily as an add-on.

Does not rely on attack-specific properties.

- Generalizes to unknown attacks.
Possible inputs take up dense sample space. But inputs we care about lie on a low dimensional manifold.
Our hypothesis for adversarial examples

Some adversarial examples are far away from the manifold. Classifiers are not trained to work on these inputs.
Our hypothesis for adversarial examples

Other adversarial example are close to the manifold boundary where the classifier generalizes poorly.
Sanitize your inputs.
Our solution

**Detector**: Decides if the example is far from the manifold.
Our solution

Reformer: Draws the example towards the manifold.
Workflow

example $x$ → Detect
Is $x$ adversarial for any detector?

Yes → $x$ is adversarial
MagNet rejects the input

No → $x$ → Reform → $x'$ → Target Classifier → class label $y$
MagNet returns $y$
Autoencoder

- Neural nets.
- Learn to copy input to output.
- Trained with constraints.

Reconstruction error:

$$\|x - ae(x)\|_2$$
Autoencoders - learn to map inputs towards manifold. - approximate input-manifold distance with reconstruction error.

Train autoencoders on **normal examples only** as building blocks.
Detector
-- based on reconstruction error

\[ \|x - x'\|_2 < \text{threshold?} \]

- **x**
- **x'**
- **x'**

Input is normal. MagNet accepts the input.

Input is adversarial. MagNet rejects the input.
Detector -- based on probability divergence

\[ D_{KL}(P||Q) \]

Input is normal. MagNet accepts the input.

Input is adversarial. MagNet rejects the input.
MagNet returns Q as final classification result.
Threat model

- blackbox defense: knows the parameters of ...
- whitebox defense: knows the parameters of ...

<table>
<thead>
<tr>
<th>Target Classifier</th>
<th>Defense</th>
</tr>
</thead>
<tbody>
<tr>
<td>blackbox defense</td>
<td>✓</td>
</tr>
<tr>
<td>whitebox defense</td>
<td>✓</td>
</tr>
</tbody>
</table>
# Blackbox Defense on MNIST Dataset

**Accuracy** on adversarial examples

<table>
<thead>
<tr>
<th>Attack</th>
<th>Norm</th>
<th>Parameter</th>
<th>No Defense</th>
<th>With Defense</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSM</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.005$</td>
<td>96.8%</td>
<td>100.0%</td>
</tr>
<tr>
<td>FGSM</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.010$</td>
<td>91.1%</td>
<td>100.0%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.005$</td>
<td>95.2%</td>
<td>100.0%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.010$</td>
<td>72.0%</td>
<td>100.0%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^2$</td>
<td>$\epsilon = 0.5$</td>
<td>86.7%</td>
<td>99.2%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^2$</td>
<td>$\epsilon = 1.0$</td>
<td>76.6%</td>
<td>100.0%</td>
</tr>
<tr>
<td>Deepfool</td>
<td>$L^\infty$</td>
<td></td>
<td>19.1%</td>
<td>99.4%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^2$</td>
<td></td>
<td>0.0%</td>
<td>99.5%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^\infty$</td>
<td></td>
<td>0.0%</td>
<td>99.8%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^0$</td>
<td></td>
<td>0.0%</td>
<td>92.0%</td>
</tr>
</tbody>
</table>
Blackbox defense on CIFAR-10 dataset

**accuracy** on adversarial examples

<table>
<thead>
<tr>
<th>Attack</th>
<th>Norm</th>
<th>Parameter</th>
<th>No Defense</th>
<th>With Defense</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSM</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.025$</td>
<td>46.0%</td>
<td>99.9%</td>
</tr>
<tr>
<td>FGSM</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.050$</td>
<td>40.5%</td>
<td>100.0%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.010$</td>
<td>28.6%</td>
<td>96.0%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^\infty$</td>
<td>$\epsilon = 0.025$</td>
<td>11.1%</td>
<td>99.9%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^2$</td>
<td>$\epsilon = 0.25$</td>
<td>18.4%</td>
<td>76.3%</td>
</tr>
<tr>
<td>Iterative</td>
<td>$L^2$</td>
<td>$\epsilon = 0.50$</td>
<td>6.6%</td>
<td>83.3%</td>
</tr>
<tr>
<td>Deepfool</td>
<td>$L^\infty$</td>
<td></td>
<td>4.5%</td>
<td>93.4%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^2$</td>
<td></td>
<td>0.0%</td>
<td>93.7%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^\infty$</td>
<td></td>
<td>0.0%</td>
<td>83.0%</td>
</tr>
<tr>
<td>Carlini</td>
<td>$L^0$</td>
<td></td>
<td>0.0%</td>
<td>77.5%</td>
</tr>
</tbody>
</table>
Detector vs. reformer

\[
\begin{align*}
\text{minimize} & \quad \|\delta\|_2 + c \cdot f(x + \delta) \\
\text{such that} & \quad x + \delta \in [0, 1]^n \\
f(x') &= \max(Z(x')_{l_x}) - \max\{Z(x')_i : i \neq l_x\}, -\kappa
\end{align*}
\]

Detector and reformer complement each other.

small distortion less noticeable

large distortion more transferable

Confidence in Carlini L² attack (distortion)

0% 20% 40% 60% 80% 100%

No defense

With detector

With reformer

With detector & reformer

reformer

no defense
Whitebox defense is not practical

To defeat whitebox attacker, defender has to either
- make it **impossible** for attacker to find adversarial examples,
- or create a **perfect** classification network.
Graybox model

- Attacker knows possible defenses.
- Exact defense is only known at run time.

Defense strategy
- Train diverse defenses.
- Randomly pick one for each session.
Train diverse defenses

With MagNet, this means training diverse autoencoders.

Our Method:
Train $n$ autoencoders at the same time.

Minimize

$$L(x) = \sum_{i=1}^{n} \text{MSE}(x, ae_i(x)) - \alpha \sum_{i=1}^{n} \text{MSE}(ae_i(x), \frac{1}{n} \sum_{j=1}^{n} ae_j(x))$$

- **reconstruction error**
- **average reconstructed image**
- **autoencoder diversity**
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.0</td>
<td>92.8</td>
<td>92.5</td>
<td>93.1</td>
<td>91.8</td>
<td>91.8</td>
<td>92.5</td>
<td>93.6</td>
</tr>
<tr>
<td>B</td>
<td>92.1</td>
<td>0.0</td>
<td>92.0</td>
<td>92.5</td>
<td>91.4</td>
<td>92.5</td>
<td>91.3</td>
<td>92.5</td>
</tr>
<tr>
<td>C</td>
<td>93.2</td>
<td>93.8</td>
<td>0.0</td>
<td>92.8</td>
<td>93.3</td>
<td>94.1</td>
<td>92.7</td>
<td>93.6</td>
</tr>
<tr>
<td>D</td>
<td>92.8</td>
<td>92.2</td>
<td>91.3</td>
<td>0.0</td>
<td>91.7</td>
<td>92.8</td>
<td>91.2</td>
<td>93.9</td>
</tr>
<tr>
<td>E</td>
<td>93.3</td>
<td>94.0</td>
<td>93.4</td>
<td>93.2</td>
<td>0.0</td>
<td>93.4</td>
<td>91.0</td>
<td>92.8</td>
</tr>
<tr>
<td>F</td>
<td>92.8</td>
<td>93.1</td>
<td>93.2</td>
<td>93.6</td>
<td>92.2</td>
<td>0.0</td>
<td>92.8</td>
<td>93.8</td>
</tr>
<tr>
<td>G</td>
<td>92.5</td>
<td>93.1</td>
<td>92.0</td>
<td>92.2</td>
<td>90.5</td>
<td>93.5</td>
<td>0.1</td>
<td>93.4</td>
</tr>
<tr>
<td>H</td>
<td>92.3</td>
<td>92.0</td>
<td>91.8</td>
<td>92.6</td>
<td>91.4</td>
<td>92.3</td>
<td>92.4</td>
<td>0.0</td>
</tr>
<tr>
<td>Random</td>
<td>81.1</td>
<td>81.4</td>
<td>80.8</td>
<td>81.3</td>
<td>80.3</td>
<td>81.3</td>
<td>80.5</td>
<td>81.7</td>
</tr>
</tbody>
</table>

Graybox classification accuracy

generate attack on

defend with
Limitations

The effectiveness of MagNet depends on assumptions that
- detector and reformer functions exist.
- we can approximate them with autoencoders.

We show empirically that these assumptions are likely correct.
Conclusion

We propose MagNet framework:

- **Detector** detects examples far from the manifold
- **Reformer** moves examples closer to the manifold

We demonstrated effective defense against adversarial examples in blackbox scenario with MagNet.

Instead of whitebox model, we advocate **graybox** model, where security rests on model diversity.
Thanks & Questions?

Find more about MagNet:

  Paper
- [https://github.com/Trevillie/MagNet](https://github.com/Trevillie/MagNet)  
  Demo code
- [mengdy.me](https://mengdy.me)  
  Author homepage